UNIVERSIDADE FEDERAL DO CEARÁ

CENTRO DE TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

NÍCOLAS DE ARAÚJO MOREIRA

Proposta de um *Front-End* em Java para Sintetizador de Voz Baseado no MBROLA

FORTALEZA

2015

NÍCOLAS DE ARAÚJO MOREIRA

Proposta de um *Front-End* em Java para Sintetizador de Voz Baseado no MBROLA

Dissertação apresentada ao PPGETI - Programa de Pós-Graduação em Engenharia de Teleinformática da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Engenharia de Teleinformática. Área de concentração: Sinais e Sistemas.

Orientador: Prof. Dr. Paulo Cesar Cortez.

FORTALEZA

2015

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca de Pós-Graduação em Engenharia - BPGE

M838p Moreira, Nícolas de Araújo.
Proposta de um Front-end em Java para sintetizador de voz baseado no MBROLA / Nícolas de Araújo Moreira. – 2015.
205 f. : il. color. , enc. ; 30 cm.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Tecnologia, Departamento de Engenharia de Teleinformática, Programa de Pós-Graduação em Engenharia de Teleinformática, Fortaleza, 2015.
Área de concentração: Sinais e Sistemas.
Orientação: Prof. Dr. Paulo César Cortez.

1. Teleinformática. 2. Inclusão digital. 3. Acessibilidade. 4. Voz - Síntese. I. Título.

CDD 621.38

UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

CAMPUS DO PICI, CAIXA POSTAL 6007 CEP 60.738-640
FORTALEZA – CEARÁ - BRASIL
FONE (+55) 85 3366-9467 – FAX (+55) 85 3366-9468

NÍCOLAS DE ARAÚJO MOREIRA

PROPOSTA DE UM FRONT-END EM JAVA PARA SINTETIZADOR DE VOZ BASEADO NO MBROLA

Dissertação submetida à Coordenação do Programa de Pós-Graduação em Engenharia de Teleinformática, da Universidade Federal do Ceará, como requisito parcial para a obtenção do grau de Mestre em Engenharia de Teleinformática.
Área de concentração: Sinais e Sistemas.

Aprovada em: 02/09/2015.

BANCA EXAMINADORA

Prof. Dr. PAULO CESAR CORTEZ (Orientador)
Universidade Federal do Ceará

Prof. Dr. DANIELO GONCALVES GOMES
Universidade Federal do Ceará

Prof. Dr. JOSÉ MARQUES SOARES
Universidade Federal do Ceará

Profa. Dra. MARIA ELIZABETH SUCUPIRA FURTADO
Universidade de Fortaleza

A Deus.

Aos meus Pais, Professores e Amigos.

A Sebastião de Araújo (*In memorian*).

A Alexandre M. de Morais (*In memorian*)

AGRADECIMENTO

À CAPES, pelo apoio financeiro com a manutenção da bolsa de auxílio, bem como à Siemens / Unify.

Ao Prof. Dr. Paulo Cesar Cortez, pela excelente orientação e aos professores participantes da banca examinadora Prof. Dr. José Marques Soares, Prof. Dr. Danielo Gonçalves Gomes e Profa. Dra. Maria Elizabeth Sucupira Furtado pelo tempo, pelas valiosas colaborações e sugestões.

Aos professores do Programa de Pós-Graduação em Engenharia de Teleinformática: Dr. Carlos Estêvão Rolim Fernandes, Dr. Charles Casimiro Cavalcante, Dr. Guilherme de Alencar Barreto, Dr. João Cesar Moura Motta, Prof. Dr. Tarcisio Marciel. À Secretaria de Acessibilidade da Universidade Federal do Ceará, pelo auxílio nos testes em campo.

Aos meus excelentes colegas e amigos do Laboratório de Engenharia de Sistemas de Computação e Siemens / Unify: Prof. MSc. Eng. Alexandre Augusto da Penha Coelho, Cincinato Furtado, Eng. Fábio Ribeiro, Prof. Dr. Helano de Souza Castro, Eng. Jacques Henrique Bessa, Eng. Jefferson Figueiredo, Luan Pinheiro, Eng. Ridley Gadelha, Tiago Gomes, Victor Fernandes, Eng. Eduardo Gabriel Bregant e Eng. Henrique Ashihara.

Aos meus grandes amigos que estiveram juntos nessa mesma caminhada e que sem a ajuda, teria sido impossível chegar aqui: Antônio Alencar, David Coelho, Delano Klinger, Ednardo Rodrigues, Franco Marques Pilloto, Henriques Zacarias, Igor Osterno, José Wilker Lima, Keuliane Nogueira, Luiza Helena Félix, Marcelo Marques Simões de Souza, Mairton Barros Junior, Marciel Barros, Reda Belkebir Mrani, Régia Talina Araújo. Meu muito obrigado pela ajuda e pelo companheirismo! E a Germano Fronza pelo esclarecimento de dúvidas.

Aos meus irmãos de Dharma do Shiwa Gonpa Guru Ling e em especial ao Lama Chimed Rigdzin. Aos amigos do Waai Dojo, em especial ao Sensei Sebastien Forêt.

Aos meus tios Tamara Duarte de Araújo e Hindemburgo Duarte de Araújo pelo apoio e suporte e à Camila Vasconcelos pela paciência e compreensão nos momentos de privação. Obrigado aos meus pais, Raimundo Moreira Filho e Samara Duarte de Araújo Moreira, pelo apoio durante mais essa longa e dura jornada.

"Quem vence alguém é um vencedor, mas quem vence a si mesmo é invencível."

- Morihei Ueshiba (Criador do Aikido)

RESUMO

Estima-se que, no Brasil, cerca de 3,46% da população apresenta grande limitação de visão e 1,6% seja totalmente incapaz de enxergar. A falta de meios de inclusão adequados impõe uma série de restrições na vida destas pessoas, em outras palavras, ferramentas de *hardware* e *software* não acessíveis geram impacto negativo na vida acadêmica, pessoal e profissional. Dentro desse contexto, a presente Dissertação tem por objetivo principal desenvolver um sistema para inclusão digital de deficientes visuais. O sistema é composto por um *front-end* multiplataforma para o sintetizador de voz MBROLA e um conjunto programas acessíveis, que inclui editor de texto, cliente de *chat*, lente de aumento virtual, entre outros, desenvolvido em Java a fim gerar um *software* multiplataforma. Além disso, o sistema é gratuito e livre para que possa atingir o maior número de usuários possível e ser modificado e aprimorado pela comunidade. A solução desenvolvida foi testada em campo, apresentando índice de inteligibilidade médio de 79% e com naturalidade classificada como razoável em um grupo de 20 usuários. Por fim, o sistema se mostrou viável, vindo a preencher uma lacuna existente no mercado brasileiro de *softwares*, permitindo maior inclusão dos deficientes visuais aos meios digitais.

Palavras-chave: Síntese de voz, Sistemas multiplataforma, Inclusão digital e acessibilidade, Deficiência visual.

ABSTRACT

It is estimated that, in Brazil, about 3.46% of population presents difficulty to see and 1.6% is blind. The lack of adequate inclusive tools imposes many restrictions on the life of these people, in other words, non-accessible hardware and software create a negative impact on academic, professional and personal life. In this context, the present thesis aims to develop a an accessible system for digital inclusion of blind users, since the existing systems present many disadvantages as low quality or cost that make impossible the daily use. The system is composed by a multiplatform Java front-end. In addition, the system is free to reach the maximum numbers of users as possible and to be modified and improved by the community. The developed solution was tested, presenting a medium intelligibility rate of 79% and naturalness classified as "reasonable" by a group of 20 users. In the end, the system proved to be feasible, filling an existing gap on Brazilian software marked, allowing greater inclusion of blind users to digital resources.

Keywords: Voice synthesis, Multiplatform systems, Digital inclusion and accessibility, Visual disability.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ACELP - *Algebric Code Excited Linear Prediction*

ADRIANE - *Audio Desktop Reference Implementation and Networking Environment*

AMR-WB - *Adaptative Multi-Rate Wideband*

API - *Application Programming Interface*

AT-SPI - *Assistive Technology – Service Provider Interface*

CDC - *Context-Dependent-Culstering*

CSS - *Concatenative Speech Syntehsis*

DAM - *Diagnostic Acceptrability Measure*

DRT - *Diagnostic Rhyme Test*

ECI - *Eloquence Command Interface*

FFT - *Fast Fourier Transform*

HMM - *Hidden Markov Models*

HMMSS - *Hidden Markov Model based Speech System*

HNM - *Harmonic-plus-Noise Model*

JSAPI - *Java Speech API*

JSML - *Java Speech Markup Language*

JVM - *Java Virtual Machine*

LPC - *Linear Predictive Coding*

MBROLA - *Multi Band Resynthesis OverLap Add*

MFCC - *Mel Frequency Cepstral Coefficients*

MLLT - *Maximum Likelihood Linear Transformation*

MOS - *Mean Opinion Score*

MRTD - *Modified Restricted Second Order TD*

NLP - *Natural Language Processor*

OMS - *Organização Mundial de Saúde*

PSOLA - *Pitch Synchronous Overlap and Add*

SAPI - *Microsoft Speech Application Programming Interface*

SMG - *Stochastic Markov Graphs*

SPR – *Symbolic Phonetic Representation*

SSML - *Speech Syntehsis Markup Language*

STFT - *Short Time Fourier Transform*

STM - *Spectral Transition Measure*

TD - *Temporal Decomposition*

TD-PSOLA - *Time Domain Pitch-Synchronous Overlap Add*

TE - *Tree Expansion*

TTS - *Text-to-Speech*

WAR - *Word Accuracy Rate*

WER - *Word Error Rate*

SUMÁRIO

1. INTRODUÇÃO

Segundo a Organização Mundial de Saúde (OMS), em 2013 existiam aproximadamente 39 milhões de pessoas com deficiência visual, outros 246 milhões sofrendo de perda moderada ou severa de visão, nas quais 90% dessas pessoas habitam em países em desenvolvimento. Esta organização calcula que 19 milhões de crianças com menos de 15 anos tenham problemas visuais. Desse total, 12 milhões sofrem de condições que poderiam ser facilmente diagnosticadas e corrigidas. Cita ainda que quase 1,5 milhão de menores têm o que é chamado de cegueira irreversível, e nunca mais voltarão a enxergar. A OMS diz que dois terços dessas crianças morrem até dois anos depois de ter perdido a visão (NAÇÕES UNIDAS DO BRASIL, 2014).

Conforme consta na Cartilha do Censo 2010, a Secretaria de Direitos Humanos da Presidência da República afirma, sobre pessoas com deficiência, que 18,6% da população brasileira apresenta deficiência visual em algum grau, sendo 3,46% severa e 1,6% totalmente deficientes. Em valores absolutos, isso significa que 6.782.860 brasileiros apresentam grande dificuldade para enxergar ou não enxergam absolutamente nada (SECRETARIA DE DIREITOS HUMANOS DA PRESIDÊNCIA DA REPÚBLICA, 2012; INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA, 2014).

A Figura 1.1 mostra os resultados obtidos no Censo de 2010 e descreve o percentual da população de 10 anos ou mais de idade, por sexo e tipo de deficiência que possui alguma ocupação, ou seja, que estuda ou trabalha. Embora se perceba que os deficientes visuais, de ambos os sexos, são os que apresentem maior ocupação dentro do grupo das pessoas com alguma deficiência, é possível ver que a porcentagem se encontra apenas um pouco acima da metade, o que indica necessidade de continuar investindo em medidas que facilitem a integração tanto social, como tecnológica e no mercado de trabalho dessas pessoas.

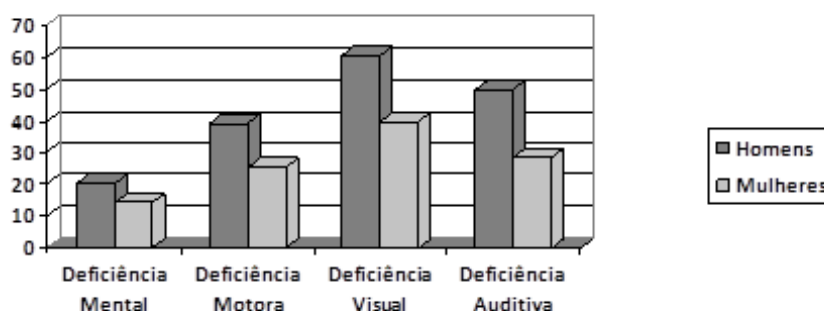



Figura 1.1: nível de ocupação da população deficiente de 10 anos ou mais de idade. Fonte: Secretaria de Direitos Humanos da Presidência da República.

1.1 Impactos da falta de acessibilidade na vida diária do deficiente visual

A partir dos dados apresentados, está evidente a imensa quantidade de pessoas com alguma deficiência, em especial, deficiência visual. Tais pessoas estão inseridas em um contexto em que se encontram mais e mais dependentes da informação nas suas atividades diárias, sendo tal fenômeno chamado de sociedade da informação. Em outras palavras: a informação é, atualmente, o item mais importante para o desenvolvimento social-político-econômico de um país (SANTOS, 2010).

De acordo com o artigo 208 da constituição federal, é dever do Estado com a educação, a garantia de atendimento educacional especializado aos portadores de deficiência, preferencialmente na rede regular de ensino. Entretanto, estudos apontam que não tem sido fornecida infraestrutura adequada o suficiente que garanta, por exemplo, o correto aprendizado por parte dos deficientes visuais: a escolaridade média das pessoas com deficiência é um ano menor que do grupo de pessoas sem deficiência, consequência da não inserção destes na escola ou da evasão. As taxas daqueles que nunca frequentaram a escola são 16,3%, 21,6% e 33,7% para a população em geral, para grupo de pessoas com visão limitada e para pessoas com total incapacidade de enxergar, respectivamente. Um estudo realizado entre estudantes a partir de 12 anos, com 26 alunos - 46,2% com visão subnormal e 53,8% com cegueira, com idade média de 17,1 anos da rede pública de ensino do Estado de São Paulo revelou que mostrou que 73,1% dos estudantes repetiram o ano. Entre as dificuldades encontradas, sobressaíram-se a leitura de livros didáticos e, dentre os que possuem visão subnormal, a dificuldade para visualizar a lousa (BRASIL, 2015; MONTILHA, 2009).

Nesse contexto, *softwares* de acessibilidade inadequados para deficientes visuais não apenas tornam espaços de trabalho ineficientes e frustrantes como também perdem

muita produtividade por subestimarem as capacidades dos funcionários. Para pessoas com deficiência, os resultados podem ser ainda piores, como dificuldade para se inserir no mercado de trabalho e dificuldade de aprendizagem. Um deficiente visual que use *hardware* e *software* apropriados consegue trabalhar pelo menos tão rápido quanto ou às vezes até mais rápido do que alguém sem deficiência visual, entretanto, a maioria dos *softwares* e sistemas operacionais permanece não acessível a este tipo de usuário e quando existem são disponibilizados em inglês (SANTOS, 2010).

Em um ambiente escolar, por exemplo, quando são detectados alunos que apresentem algum grau de redução visual, a conduta mais indicada deve ser, sempre, no sentido de buscar e garantir os recursos didáticos e pedagógicos que melhor atendam às necessidades destes indivíduos.

A educação especial visa desenvolver tecnologias de *hardware* e *software*, adaptando-os para auxiliar na solução do problema do processo de aprendizagem de pessoas que não possuem o seu desenvolvimento cognitivo normal, tais como os deficientes visuais, entre outros. Através da exploração dos recursos das novas tecnologias da informação é possível criar ambientes de aprendizagem, visando o desenvolvimento cognitivo dos portadores de necessidades especiais.

Diante deste problema, a interface do *software* educacional deve ser projetada de forma a melhor responder às necessidades do usuário. Com relação aos deficientes visuais, destacam-se alguns requisitos que devem ser atendidos pela interface, tais como a utilização de sons para interação usuário-máquina e privilegiando o uso do teclado através de teclas de atalho, evitando mensagens visuais e interação através do mouse (SUN MICROSYSTEMS, 1998).

Considere alguém em idade economicamente ativa e que sofreu uma perda da visão. Tarefas como ler um jornal, parte do ritual matinal, são impossíveis sem auxílio de uma ferramenta de inclusão de deficientes visuais. A pessoa não pode mais ver as horas no mostrador de um relógio digital ou ajustar o alarme do mesmo sem auxílio. Também não pode ler *e-mails*, *fax*, correspondências sem assistência. Não é possível reconhecer o rosto das pessoas com quem convive e muitos equipamentos se tornam impossíveis de serem usados porque simplesmente o projeto assume que todos os usuários possuem as mesmas habilidades.

As barreiras encontradas pelos usuários portadores de deficiência visual afetam áreas como emprego, educação e a possibilidade de uma vida independente. Se uma pessoa não consegue usar um telefone, atividades são severamente restritas porque até

mesmo comunicações básicas se tornam difíceis. Se uma pessoa não consegue usar um computador, conseguir uma vaga de emprego ou frequentar uma universidade se tornam atividades desafiadoras, ou até talvez impossíveis. Se as pessoas desejam ter acesso à internet, mas não conseguem ler o conteúdo das páginas, não têm acesso ao comércio eletrônico, informações básicas e até mesmo interações sociais. Até mesmo a privacidade fica afetada, uma vez que deverão delegar a outras pessoas tarefas de natureza pessoal, como ler e-mails. Em resumo, a falta de acessibilidade exclui do portador de deficiência visual independência e liberdade (SUN MICROSYSTEMS, 2003).

Quando alguém adquire uma deficiência, seja física, sensorial, cognitiva ou outra, não é somente a habilidade reduzida que é afetada. É necessário começar a atuar em um mundo em que muitos aspectos da vida diária mudam radicalmente.

1.2 Soluções existentes para a integração social dos deficientes visuais no Brasil

Diversas medidas têm sido adotadas por governos e pela sociedade a fim de assegurar a integração social de pessoas com deficiência. Deficientes auditivos podem, por exemplo, acompanhar programação televisionada por meio de funções como *Closed Caption* - legendas que transcrevem o que está sendo falado. Instituições de ensino tem procurado difundir a Linguagem Brasileira de Sinais (LIBRAS) a fim de melhor capacitar ouvintes e deficientes auditivos, permitindo que ambos possam se comunicar por meio de uma linguagem padrão. Nos setores da construção civil, arquitetura e transporte, tem se difundido os conceitos de acessibilidade por meio da construção de rampas e elevadores para garantir uma melhor mobilidade daqueles que apresentam dificuldade de locomoção.

No caso de deficientes visuais, bibliotecas públicas têm procurado disponibilizar parte de seu acervo em Braille. Além disso, medidas de acessibilidade para deficientes visuais têm sido preocupações não só a nível nacional como também internacional. Organismos responsáveis por estabelecer padrões na internet teceram recomendações e normas a fim de assegurar o acesso para aqueles que enxergam com dificuldade, incluindo formas de modificar o tamanho da fonte e alto contraste em sites.

Fazer o computador pronunciar uma linha de texto e exibir o texto num dispositivo Braille são as formas mais comuns para cegos aprenderem o que está escrito na tela do computador. Um dos dispositivos Braille mais comuns é denominado de linha e consiste

de uma tela tátil com seis ou oito pontos por letras que pode ser lido por meio do toque pelos usuários que conhecem o alfabeto Braille. (KNOPPER, 2009).

A Figura 1.2 mostra o alfabeto Braille para português. Cada idioma usa uma tabela diferente para tradução e, como não há símbolos especiais para números, as letras de “a” a “j” são usadas para representar os algarismos de um a zero, às vezes com um “símbolo de número” antes para esclarecer que se tratam de dígitos.

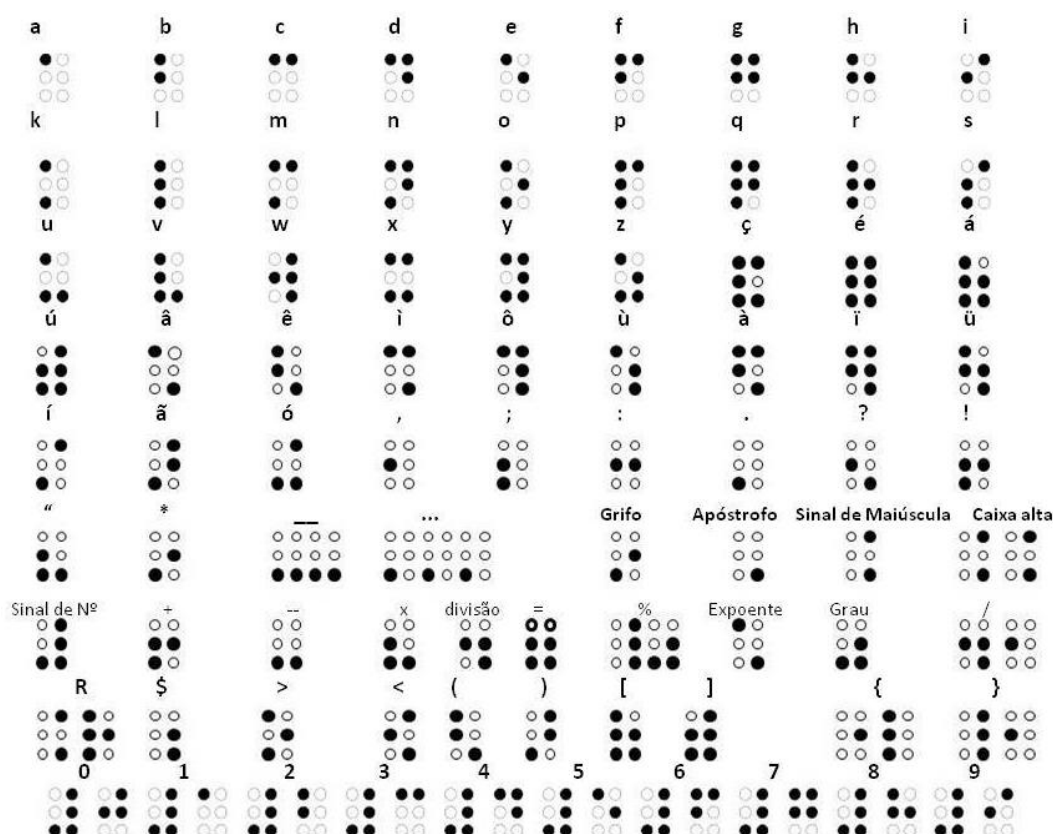



Figura 1.2: alfabeto Braille para português. Fonte: Google Images.

Sistemas TTS, *Text to Speech*, são sistemas que transformam um texto simples em voz falada, sendo, atualmente, importantes ferramentas para a interação homem-computador, podendo ser utilizados como leitores de tela para deficientes visuais (COSTA e MONTE, 2012).

Dentre os sistemas de acessibilidade e/ou síntese de voz existentes atualmente, podem-se destacar como principais, os seguintes: ADRIANE, Acapella, DOSVOX, eSPEAK, Festival, Jaws, LianeTTS, MBROLA e Virtual Vision.

Embora estes sistemas apresentem desempenho considerado adequado aos seus propósitos, possuem características que, de uma forma ou de outra, limitam o seu uso a

um grupo menor de usuários, seja por serem exclusivos para uma determinada plataforma operacional, como ADRIANE, exclusivo para GNU/Linux, ou simplesmente não serem nativamente multiplataformas - neste caso, todos esses citados, não possuir suporte à língua portuguesa, como no caso do Festival, não serem gratuitos ou não serem livres, como Acapella, Virtual Vision e Jaws, por exemplo. Adicionalmente, tais sistemas ainda não serem um sistema TTS completo, como o MBROLA, ou se basearem em vozes pré-gravadas, como o DOSVOX, por exemplo, o que limita as possibilidades de interação.

Diante deste cenário, percebe-se a necessidade de um sistema que garanta aos deficientes visuais amplo acesso aos recursos oferecidos pela informática, que gere maior impacto na integração social desse grupo, ou em outras palavras, que promova de fato a inserção digital por meio de um pacote de *softwares.* Neste caso, que inclui não somente um sintetizador de voz, mas também um editor de texto, cliente de e-mails e chat, lente de aumento, etc. Todas estas ferramentas voltadas para usuários com problemas visuais em seus mais diversos graus - desde a dificuldade de enxergar a curta distância, que exigiria o uso de uma lente de aumento virtual, até cegueira plena, por meio do uso de um sintetizador de voz para que seja possível a interação do usuário com o computador.

1.3 Objetivos

1.3.1 Geral

Esta Dissertação tem por objetivo principal propor um *Front-End* em Java para Sintetizador de Voz Baseado no MBROLA a fim de promover a inclusão digital de deficientes visuais.

1.3.2 Específicos

Durante o desenvolvimento desta Dissertação, outros objetivos foram atingidos, tais como: teste de usabilidade dos principais sintetizadores de voz existentes, como Acapella, DOSVOX / LINVOX, eSpeak, FreeTTS, Furbspeech, IBM Via Voice e JSAPI testes de naturalidade, inteligibilidade e usabilidade envolvendo usuários deficientes visuais para fim de validação do sistema desenvolvido e estudos sobre uso de filtros digitais em lentes de aumento virtuais. Além disso, como objetivo secundário, desenvolveu-se um *front-end* flexível o suficiente para suportar o sintetizador de voz do Google Speech API e FreeTTS, ou seja, um sistema que atua como interface entre de texto de entrada e tais sintetizadores, fornecendo um formato intermediário apropriado para os mesmos.

Outrossim, o presente trabalho pretende servir de referência bibliográfica sobre o tema, apresentando conceitos referentes à acessibilidade, anatomia, fisiologia e dinâmica do trato vocal, fonética, modelos e algoritmos de síntese de voz bem como as soluções existentes no mercado, fazendo uma análise completa e bastante aprofundada a respeito destes temas.

1.4 Trabalhos aceitos em congressos relacionados

MOREIRA, Nícolas de Araújo e CORTEZ, Paulo Cesar. *Protótipo de Sistema de Acessibilidade e Síntese de Voz Livre e Multiplataforma*. In: INFOBRASIL 2014. 2014. Fortaleza, Brasil.

MOREIRA, Nícolas de Araújo e CORTEZ, Paulo Cesar. *A Multiplaftorm and Open Source Accessibility System for Portuguese Language*. In: World Conference on Information Systems and Technologies - WordCIST 2015. 2015. Azores, Portugal.

1.5 Estruturação do trabalho

O Capítulo 2 (Conceitos básicos e fundamentos) introduz todos os conceitos básicos relacionados à natureza do trabalho, apresentando algumas definições referente à acessibilidade, anatomia, fisiologia e dinâmica do trato vocal, noções básicas de fonética da língua portuguesa e características da voz humana.

O Capítulo 3 (Visão geral e projeto de um sistema de síntese de voz via *software*: aspectos qualitativos e problemas relativos) apresenta a visão geral de um sistema de síntese de voz, citando seus componentes e funcionamento global. Ademais, o método de síntese de voz baseada em concatenação é explicado de forma detalhada, bem como são discutidos os aspectos qualitativos e problemas relativos à síntese de voz apresentados pelos sistemas disponíveis atualmente.

O Capítulo 4 (Tecnologias de síntese de voz e acessibilidade existentes no mercado e o MBROLA) cita os principais sistemas de síntese de voz e acessibilidade existentes no mercado, desde trabalhos acadêmicos até produtos já consagrados, passando por sistemas que não se encontram mais disponíveis, citando suas características, vantagens e desvantagens. Ademais, apresenta de forma detalhada o MBROLA, sistema em que se baseia o presente trabalho.

O Capítulo 5 (Metodologia) explana a metodologia desenvolvimento da solução proposta, apresentando as ferramentas utilizadas para o seu desenvolvimento e suas características.

O Capítulo 6 (Testes e resultados obtidos), explica a metodologia de teste utilizada, mostrando os resultados obtidos e comparando suas vantagens sobre as ferramentas existentes atualmente (apresentados no Capítulo 4). São apresentados resultados tanto qualitativos como quantitativos, apresentando depoimentos de usuários do sistema bem como comparando com trechos de vozes naturais.

O Capítulo 7 (Conclusão) encerra o presente trabalho com uma visão geral a respeito do tema abordado bem como apresenta as dificuldades encontradas e sugere melhorias, implementação de novos recursos, citando trabalhos complementares que possam colaborar com a melhoria do projeto.

Além disso, o presente trabalho conta com dois importantes apêndices sobre modelagem matemática do trato vocal e algoritmos de síntese de voz, que apresentam as principais técnicas de síntese de voz, descrevendo seus modelos e algoritmos, citando suas vantagens e desvantagens e comparando os resultados de cada um, desde as primeiras implementações até os trabalhos mais recentes em cada abordagem, ou seja, das técnicas clássicas até o estado da arte. Embora envolva métodos não relacionados ao trabalho diretamente, estes apêndices objetivam serem referências para outros trabalhos na área e apresentam um comparativo entre as técnicas de síntese de voz, realizando uma revisão bibliográfica que serve de embasamento teórico para a escolha da técnica de síntese concatenativa como *back-end* do projeto.

2. FUNDAMENTOS E CONCEITOS BÁSICOS

Este Capítulo visa apresentar os conceitos básicos relacionados à temática da presente Dissertação, cobrindo conceitos relacionados à acessibilidade, anatomia, fisiologia e dinâmica do trato vocal, noções básicas de fonética da língua portuguesa e características da voz humana.

2.1 Deficiência visual

A classificação entre os grupos de deficiência visual, cegos e portadores de visão subnormal se dá por meio de duas escalas oftalmológicas: acuidade visual e o campo visual. O primeiro se refere à capacidade de se enxergar a uma determinada distância. O segundo se refere à amplitude da área alcançada pela visão (INSTITUTO BENJAMIN CONSTANT, 2014).

O termo cegueira não significa necessariamente a total incapacidade de ver, mas sim o prejuízo dessa capacidade para o exercício de tarefas rotineiras. Denomina-se cegueira parcial, também chamada de cegueira legal ou cegueira profissional, aquela cujos indivíduos são capazes apenas de contar os dedos a curta distância e que percebem apenas vultos. A cegueira total é a completa perda de visão, chamada de visão nula, a qual não há sequer percepção luminosa, sendo chamada pelos oftalmologistas de visão zero (INSTITUTO BENJAMIN CONSTANT, 2015).

Pedagogicamente, define-se cego, o indivíduo que necessita de instrução em Braille, mesmo que possua visão subnormal e como portador de visão subnormal aquele que lê tipos impressos ampliados ou com auxílio de recursos ópticos mais poderosos (INSTITUTO BENJAMIN CONSTANT, 2015).

De acordo com o Instituto Benjamin Constant, uma pessoa é considerada cega se corresponde a um dos critérios seguintes: a visão corrigida do melhor dos seus olhos é de 20/200 ou menos, isto é, se ela pode ver a 20 pés (6 metros) o que uma pessoa de visão normal pode ver a 200 pés (60 metros), ou se o diâmetro mais largo do seu campo visual subentende um arco não maior de 20 graus, ainda que sua acuidade visual nesse estreito campo possa ser superior a 20/200 (INSTITUTO BENJAMIN CONSTANT, 2015).

Nesse contexto, caracteriza-se como portador de visão subnormal aquele que possui acuidade visual de 6/60 e 18/60 em escala métrica e/ou um campo visual entre 20 e 50º (INSTITUTO BENJAMIN CONSTANT, 2015).

2.2 Inclusão digital

Por inclusão digital, entende-se como a permanente busca por igualdade de condições e oportunidades a fim de evitar situações de privação. Na prática, isso significa favorecer o acesso do cidadão ao mundo virtual, reduzir o analfabetismo digital por meio do fornecimento de conhecimento básico sobre informática e melhorar e adaptar a interface para o seu público-alvo. Em outras palavras, para pessoas com necessidades especiais, a inclusão digital envolve quebrar barreiras arquitetônicas, de comunicação e de acesso físico a equipamentos e *softwares* adequados, e que são necessárias ferramentas que adaptem e adequem o equipamento de tal forma que o usuário o use satisfatoriamente (SANTOS, 2010).

Inclusão digital é um processo muito mais profundo que permitir acesso a um computador, envolvendo também capacitar o indivíduo a operar um computador com autonomia (SANTOS, 2010).

A questão da inclusão digital ganha uma dimensão ainda mais complexa quando o usuário é portador de necessidades especiais. As pessoas com deficiência passaram a receber maior atenção por meio de políticas específicas voltadas para a qualificação e a habilitação de tal forma que as capacite e as integre à sociedade. Entretanto, ainda existem barreiras físicas que dificultam o acesso do usuário ao computador e nesse caso, os obstáculos para este acesso não se restringe apenas a questões socioeconômicas, mas também questões físicas (SANTOS, 2010).

2.3 Acessibilidade

O Decreto 5.296 de 2 de dezembro de 2004 define acessibilidade como *"condição para utilização, com segurança e autonomia, total ou assistida, dos espaços, mobiliários e equipamentos urbanos, das edificações, dos serviços de transporte e dos dispositivos, sistemas e meios de comunicação e informação, por pessoa portadora de eficiência ou com mobilidade reduzida."* (SANTOS, 2010).

O conceito de acessibilidade se aplica aos sistemas de informação por meio de dispositivos eletrônicos, incluindo computadores *desktops*, *notebooks*, celulares que quebrem as barreiras, seja adaptando *hardware*, seja utilizando *softwares* apropriados ou ambos (SANTOS, 2010).

Ainda segundo (SANTOS, 2010): *"A acessibilidade digital só pode ser proporcionada por meio de combinação entre hardware e software que oferecem,*

respectivamente, mecanismos físicos para superar barreiras de percepção e o acesso a funções e informações.”

2.4 Tecnologias assistivas

Tecnologia assistiva é aquela que provê suporte a portadores de necessidades especiais, adaptando e/ou fornecendo dispositivos necessários para que essas pessoas possam realizar atividades da forma mais independente possível. Este tipo de tecnologia proporciona, às pessoas com necessidades especiais, maiores independência e qualidade de vida, refletindo-se nas relações sociais, no trabalho e também na família (SANTOS, 2010).

A tecnologia assistiva, quando corretamente aplicada, é fundamental para garantir acessibilidade às mesmas atividades realizadas pelas pessoas sem necessidades especiais. Essas técnicas podem eliminar ou minimizar as limitações funcionais, permitindo seu desempenho e interação nas mais diversas situações cotidianas, como, por exemplo, o acesso à informação e à comunicação (SANTOS, 2010).

Como a informação processada por um computador é exibida em monitores de vídeo, pessoas com deficiência visual total ou parcial precisam recorrer a outros dispositivos para obter as informações da tela. Deve-se então fornecer um *software* leitor de tela que capte a informação do vídeo e a envie para um sintetizador de voz ou para um terminal Braille (SANTOS, 2010).

Dentre as tecnologias assistivas voltadas para deficientes visuais, pode-se citar as seguintes soluções principais: sintetizador de voz - processo de produção artificial de voz humana; leitor de tela, um *software* que, com auxílio de um sintetizador de voz, transforma os textos impressos na tela em voz humana e um ampliador de tela, que funciona como uma lupa / lente de aumento, aumentando o tamanho dos itens exibidos na tela do computador (SANTOS, 2010).

Para compreender melhor tais tecnologias, é importante conhecer melhor sobre a fisiologia da voz, em especial o trato vocal.

2.5 Fonética e especificidades de cada língua

Define-se fonema como a menor unidade sonora de uma língua, assim, fonemas são as unidades sonoras básicas de uma língua (SCHROETER, 2005 ; MACHADO, 1997).

Os sons podem ser classificados em classes fonética de acordo com a forma de articulação, como, por exemplo, vogais, fricativos, pausas, nasais, deslizantes, líquidos,

ditongos, etc. Podem ser classificados também de acordo com o local da articulação: labial, dental, alveolar, palatal, velar, uvular, faríngeo e glotal. Outros tipos de classificação podem incluir sussurros, fonação respiratória, chiados, etc. (DUTOIT, 1997).

Os sons produzidos durante a fala são divididos em vozeados, aqueles em que as pregas vocais vibram durante a produção, e não vozeados, aqueles em que as pregas vocais não vibram durante a sua produção (MACHADO, 1997).

As vogais são distinguidas pela posição da língua e dos lábios e se classificam quanto à zona de articulação, região da boca em que se dá a maior elevação da língua, podendo ser anterior, central e posterior; pela elevação da região mais alta da língua, podendo ser classificadas em altas, médias e baixas; e quanto ao timbre, podendo ser aberta ou fechada (MACHADO, 1997).

A classificação das vogais da língua portuguesa, os valores das frequências dos seus três primeiros harmônicos, os formantes, em Hz, e a intensidade média dos harmônicos em dB com seus respectivos desvios-padrão, para ambos os sexos são mostrados respectivamente nas Tabelas 2.1, 2.2 e 2.3.

Tabela 2.1: classificação das vogais.

		Anteriores	**Centrais**	**Posteriores**
Altas		/i/		/u/
Médias	**Fechadas**	/e/		/o/
	Abertas	/é/		/ó/
Baixas			/a/	

Fonte: (MACHADO, 1997).

Tabela 2.2: média dos valores das frequências dos harmônicos correspondentes aos três primeiros formantes (F1, F2, F3), em Hz, para cada vogal, para ambos os sexos.

	Mulheres			**Homens**		
	F1	**F2**	**F3**	**F1**	**F2**	**F3**
/a/	1002,90	1549,95	2959,70	753,87	1278,70	2483,44
/e/	672,45	2242,93	3018,60	688,44	1745,11	2566,00
/e/	437,03	2429,76	3087,09	406,63	1955,60	2540,33
/i/	361,90	2583,89	3378,14	297,80	2150,85	2925,14
/É/	715,34	1073,27	2981,69	580,15	947,25	2525,52
/o/	444,89	914,26	2899,80	411,62	832,84	2376,13
/u/	461,82	763,41	2902,55	345,27	799,51	2351,50

Fonte: (GONÇALVES et. al., 2009).

Tabela 2.3: média dos valores das intensidades dos harmônicos (em dB) e respectivos desvios-padrão, para cada vogal, para ambos os sexos.

	Mulheres		**Homens**	
	X	**DP**	**X**	**DP**
/a/	42,92	9,48	36,91	11,02
/e/	45,04	6,88	39,37	9,27
/e/	43,88	8,32	38,85	10,3
/i/	41,49	10,7	36,73	10,97
/É/	39,94	11,56	36,33	11,63
/o/	36,5	13,25	33,84	12,91
/u/	35,29	13,56	32,78	13,02

Fonte: (GONÇALVES et. al., 2009).

As consoantes podem ser classificadas quanto ao modo de articulação - indicando o tipo de obstáculo encontrado pelo fluxo de ar ao passar pela boca, sendo oclusivas ou constritivas. Nas oclusivas, há total constrição do ar, enquanto que nas constritivas, a constrição é parcial. As constritivas se subdividem em fricativas, laterais e vibrantes. Nas fricativas, o ar sofre fricção, enquanto que nas laterais o ar passa pelos lados da cavidade bucal. Já nas vibrantes, a língua ou o véu palatino vibram.

Quanto ao ponto de articulação - indicando o ponto da cavidade bucal onde se encontra o obstáculo à corrente de ar, as consoantes podem ser classificadas em bilabiais, labiodentais, alveolares, palatais e velares. Nas bilabiais, os lábios entram em contato, enquanto que nas labiodentais o lábio inferior toca os dentes incisivos superiores. Já nas alveolares, a língua toca os alvéolos dos incisivos superiores e nas palatais a língua toca o palato duro (o “céu da boca”). Nas velares, a língua toca o palato mole – véu palatino.

As consoantes podem ainda ser classificadas de acordo com a vibração das pregas vocais, surdas ou sonoras, e ainda de acordo com a participação das cavidades bucais e nasal para a sua produção, as orais e nasais (MACHADO, 1997).

Tabela 2.4: classificação das consoantes.

Cavidades Bucal e Nasal		**Orais**						**Nasais**
Modo de Articulação		**Oclusivas**		**Constritivas**				
				Fricativas		**Laterais**	**Vibrantes**	
Pregas vocais		**Surdas**	**Sonoras**	**Surdas**	**Sonoras**	**Surdas**	**Sonoras**	**Sonoras**
Ponto de Articulação	**Bilabiais**	/p/	/b/					/m/
	Labiodentais			/f/	/v/			
	Linguodentais	/t/	/d/					/n/
	Alveolares			/s/	/z/	/l/	/r/	
	Palatais			/x/	/j/	/λ/		/η/
	Velares	/k/	/g/				/R/	

Fonte: (MACHADO, 1997).

Para a formação de fonemas existem dois conjuntos de parâmetros que determinam o som produzido: as frequências de ressonância do trato vocal, os formantes, e a frequência dos pulsos de ar produzidos pelo conjunto composto por pulmões e pregas vocais. Tais parâmetros são responsáveis tanto pela diferenciação entre fonemas quanto por locutores. Como os parâmetros que diferenciam fonemas entre si são os formantes e como o trato vocal varia de uma pessoa a outra, cada fonema possui um conjunto de formantes acrescidos dos formantes característicos de cada trato vocal (MACHADO, 1997).

Vogais diferem das consoantes de acordo com o grau de abertura do trato vocal. Se o trato vocal está aberto o suficiente para o ar pulsado pelos pulmões fluir sem encontrar obstáculos, uma vogal é produzida. A atuação da boca é então reduzida a simplesmente modificar o timbre vocal, caso contrário é produzida uma consoante (DUTOIT, 1997).

Sons vozeados, como os produzidos por uma vogal, por exemplo, ocorrem quando o ar é forçado pelos pulmões, através das pregas vocais, em direção à boca ou nariz, por onde escapa, ou seja, Sons vozeados são produzidos pela excitação do trato vocal gerado por pulsos de ar glotal quasi-periódicos resultantes da vibração das pregas vocais (LOPEZ, 2009; SPANIAS, 1994).

Quando ocorre a constrição de algum ponto do trato vocal, geralmente em direção à boca, sons fricativos ou não-vozeados são originados, forçando o ar passar pela constrição com uma velocidade suficientemente grande para gerar uma turbulência e consequentemente um ruído que excita o trato vocal. Sons fricativos incluem /ch/, /f/, /s/, /v/, /x/ e /z/, ou seja, sons não-vozeados são produzidos forçando o ar ao longo de uma constrição do trato vocal (LOPEZ, 2009; SPANIAS, 1994).

Quando há a total obstrução de algum ponto ao longo da passagem de ar no trato nasal, há produção de sons nasais, como /m/ ou /n/. A cavidade oral, embora constrita, permanece acusticamente acoplada à faringe e, dessa forma, a boca atua como cavidade ressonante, ou seja, sons nasais como, por exemplo, o /n/, se devem devido ao acoplamento acústico do trato nasal com o trato vocal (LOPEZ, 2009; SPANIAS, 1994).

Sons pulsantes, como o /p/ por exemplo, são produzidos ao soltar a pressão do ar produzido atrás do fechamento do trato vocal abruptamente (SPANIAS, 1994).

Com o intuito de uma visão global da complexidade da língua portuguesa, seus fonemas estão mostrados na Tabela 2.5.

Tabela 2.5: fonemas da língua portuguesa.

Símbolo	**Exemplo**	**Transcrição Fonológica**
/p/	**P**aca	/paka/
/b/	**B**ula	/bula/
/t/	**T**ara	/tara/
/d/	**D**ata	/data/
/k/	**C**ara, **qu**ero	/kara/,/kéro/
/g/	**G**ola, **g**uerra	/góla/, /géRa/
/f/	**F**aca	/faka/
/v/	**V**ala	/vala/
/s/	**S**ola, a**ss**a, mo**ç**a	/sola/, /asa/, /mosa/
/z/	A**s**a, **z**ero	/aza/, /zéro/
/x/	Me**ch**a, **x**á	/méxa/, /xa/
/j/	**J**aca, **g**ela	/jaka/, /jela/
/m/	**M**ola	/móla/
/n/	**N**ata	/nata/
/η/	Ni**nh**o	/niηo/
/l/	**L**ata	/lata/
/λ/	Ca**lh**a	/kaλa/
/r/	Pa**r**a	/para/
/R/	**R**ota, ca**rr**oça	/róta/,/KaRosa/
/a/	C**á**	/ka/
/é/	M**e**l	/mél/
/e/	S**e**da	/seda/
/i/	R**i**ca	/rica/
/ó/	M**o**la	/móla/

Tabela 2.5: fonemas da língua portuguesa (continuação).

Símbolo	Exemplo	Transcrição Fonológica
/o/	T**o**la	/tola/
/u/	G**u**la	/gula/

Fonte: (MACHADO, 1997).

Cada idioma possui suas especificidades com relação à sua estrutura sonora. A unidade de fala básica da língua chinesa, por exemplo, é a sílaba: as sílabas são compostas por vogais ou por uma vogal em conjunto com uma consoante. Há 414 sílabas em chinês, totalizando 1716 sílabas se incluir também tons. A língua eslovaca apresenta apenas 1550 dífonos frequentes (KANG et. Al., 2009; TALAFOVÁ et. al., 2007).

A língua japonesa apresenta maior dificuldade no tocante à análise morfológica e léxica de palavras quando comparada com línguas europeias. A maioria dos caracteres japoneses apresentam diversas pronúncias diferentes, dependendo de seu significado e contexto. A língua japonesa contém 38 fonemas básicos, entretanto, no tocante à síntese de voz, a qualidade da voz usando apenas esse limitado conjunto é bastante sofrível (KOBAYASHI et. al., 1998).

2.6 Características da voz

A voz humana pode ser caracterizada pelos seguintes atributos: tom, timbre, duração e intensidade. O tom define a altura musical da voz e pode ser classificado em agudos e graves. Vozes masculinas podem ainda ser classificadas em tenor, barítono e baixo. Para fins de acessibilidade, considera-se mais agradável o tom barítono. Já o timbre é o matiz pessoal da voz, que é um parâmetro complexo determinado pelo tom fundamental e seus harmônicos, podendo ser caracterizado como agradável, rouco, chiado, etc. Por outro lado, a duração o som é a propriedade que permite que seja classificado em curtos, longos e toda gama de intermediários, como semi-longos, semi-curtos, etc. Por fim, a intensidade é a propriedade que se refere à maior ou menor força com que se produz voz, podendo ser classificado como voz forte ou fraca.

Para uma adequada síntese de voz, é necessário equilíbrio entre tipo de voz, intensidade, velocidade, frequência, pronúncia, ressonância e articulação. Quando tais fatores não estão em equilíbrio, a voz resultante pode apresentar efeitos como rouquidão, aspereza, tensão, hipersensibilidade, entre outros (MATUCK, 2005).

2.6.1 Propriedades Matemáticas da Voz

A voz é um sinal essencialmente não estacionário, ou seja, se todas as características de seu comportamento são alteradas no tempo. Entretanto, pode-se aproximar a condição de estacionaridade ao se observar localmente o sinal de voz em janelas temporais de curta duração, tipicamente de 5 a 20ms, assim, as propriedades estatísticas e espectrais são definidos dentro destes segmentos (SPANIAS, 1994).

A voz humana pode produzir sons vozeados, como, por exemplo /a/ e /i/, e não vozeados, como o /sh/, por exemplo, que são quasi-periódicos no domínio do tempo e harmonicamente estruturados no domínio da frequência, enquanto que sons não vozeados são aleatórios. Além disso, a energia de sons vozeados é geralmente maior que a energia dos segmentos de voz não vozeados (SPANIAS, 1994).

A distribuição das frequências (espectro) da voz é caracterizada por sua estrutura harmônica e formante. A estrutura harmônica é uma consequência da quasi-periodicidade e pode ser atribuído à vibração das pregas vocais. A estrutura formante, envelope espectral, deve-se à interação entre a fonte e o trato vocal (SPANIAS, 1994).

O envelope espectral é caracterizado por um conjunto de picos chamados de formantes. Os formantes são os modos ressonantes do trato vocal. Em média, o trato vocal apresente de 3 a 5 formantes abaixo de 5 kHz. As amplitudes e localizações dos três primeiros formantes, que geralmente ocorrem abaixo de 3 kHz, são muito importantes tanto na síntese quanto na percepção. Altos formantes são também importantes para representações de sons com grande largura de banda e vozeados.

A diferença entre os espectros de sons vozeados e não vozeados está mostrada na Figura 2.1. Os dois gráficos à esquerda desta Figura mostra os sinais no domínio do tempo e os sinais à direita são os respectivos espectros de frequência, sendo que o primeiro sinal é vozeado e o segundo não vozeado. A diferença entre os dois está evidente no envelope e na magnitude dos componentes de frequência de cada um dos sinais e na periodicidade.

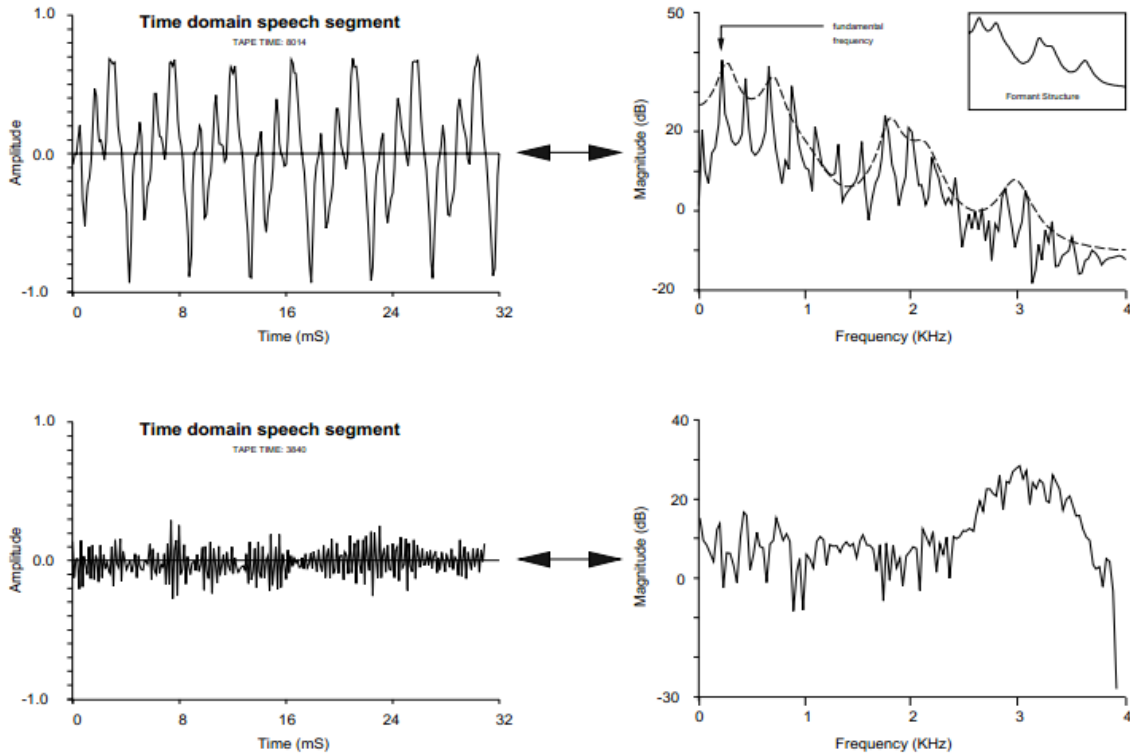



Figura 2.1: espectro de sons vozeados e sons não vozeados. Fonte: (SPANIAS, 1994).

Para efeitos de análise, o sinal é considerado nulo fora da janela de observação, o intervalo de 5 a 20 ms em que o sinal é considerado estacionário, ou seja, seja $s(n)$ sinal de voz, o sinal observado $x(n)$ é dado por $x(n) = s(n)w(n)$, em que:

$$w(n) = \begin{cases} 1, & 0 \leq n \leq N-1 \\ 0, & n < 0 \; ou \; n > N-1. \end{cases} \tag{2}$$

A transformada localizada de Fourier $X(n,\theta)$, chamada também de STFT - *Short Time Fourier Transform*, do sinal $x(n)$ é a transformada de Fourier do sinal $x(m)w(n-m)$ em que $w(n)$ é uma janela de ponderação

$$X(n,\theta) = \sum_{m=-\infty}^{\infty} x(m)w(n-m)e^{-jm\theta}, \qquad \theta = 2\pi fT \tag{3}$$

e para um sinal amostrado

$$X(n,k) = \sum_{m=0}^{N-1} x(m)w(n-m)e^{-j2\pi km/N}. \tag{4}$$

A transformada localizada de Fourier é uma função da frequência f ou k e do instante central de observação.

Devido à natureza não estacionária do sinal de voz, a transformada localizada de Fourier é uma das principais ferramentas de análise na frequência. O resultado que se obtém e a convolução da transformada do sinal com a transformada da janela.

Para análise de um segmento longo não estacionário de sinal, usa-se uma janela deslizante no tempo e para cada posição da janela, determina-se a transformada localizada.

2.6.2 Frequência Fundamental

Pitch é frequência fundamental de vibração das cordas vocais que produzem sons vozeados que é a característica mais importante no que diz respeito à capacidade de transmitir informação linguística, enquanto que a duração provê o ritmo da fala. Esta frequência também indica a proeminência de palavras importantes, por meio de subidas e descidas, em conjunto com o aumento e diminuição da duração dos segmentos Além disso, aumenta a inteligibilidade, uma vez que a variação da frequência fundamental contém informações sobre a estrutura sintática e sobre o estado psicológico (SCHROETER, 2005; AZUIRSON, 2009).

2.6.3 Timbre

O timbre, ou cor sonora, é uma qualidade auditiva por meio do qual o ser humano identifica os diversos tipos de voz, bem como instrumentos musicais e outras fontes sonoras. Tal qualidade sonora está correlacionada com a forma da onda sonora sendo que frequência e a amplitude são importantes na definição do timbre (LIMA, 2010).

Hermann Von Helmholtz, no final do século XIX, caracterizou os sons como constituído por uma forma arbitrária fechada em um envelope, envoltória, de amplitude composta por três partes: ataque, também chamado de tempo de crescimento, período estável e queda, chamado de tempo de queda. O ataque é o tempo que a amplitude de um som leva para sair do zero e subir até o valor de pico. O período estável é aquele que a amplitude é idealmente constante e o som desaparece no período de queda, em que a amplitude cai até zero (LIMA, 2010).

Um envelope de uma onda sonora está mostrado na Figura 2.2, evidenciando o intervalo de ataque, o período estável e o intervalo de queda.

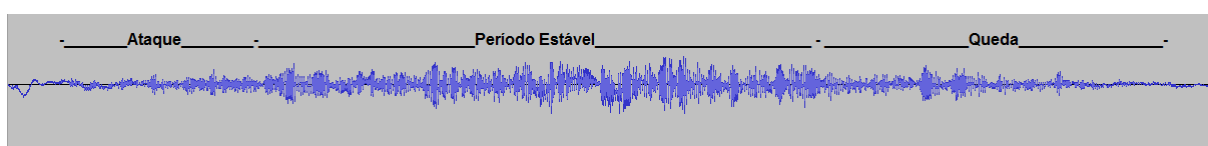



Figura 2.2: envelope de uma onda sonora. Fonte: (LIMA, 2010 - Adaptado).

O timbre é determinado pelas cavidades ósseas, cavidades nasais, boca, garganta, traqueia e pulmões, além da própria laringe (MATUCK, 2005).

2.6.4 Prosódia

A prosódia é uma interpretação rítmico-melódica da sintaxe e da semântica. Uma das funções da prosódia é fornecer indicações sobre a localização de acentos, criando uma sensação de ritmo. A prosódia determina como uma sentença é falada em termos de melodia, ritmo, sotaque e emoções e pode carregar significados até mesmo em línguas não-tonais.

A prosódia é um processo de natureza supra-segmental que atua em sílabas, palavras, orações, etc.. Os três principais parâmetros prosódicos são: duração, frequência fundamental e intensidade, sendo os dois primeiros os mais importantes. A modelagem da intensidade não produz ganhos significativos de qualidade da síntese de voz (AZUIRSON, 2009).

Uma prosódia errada pode prejudicar severamente a inteligibilidade / compreensão, assim uma modelagem adequada dos parâmetros prosódicos, duração e frequência, dos fonemas pode melhorar significativamente a inteligibilidade e a naturalidade do resultado de um sintetizador de voz. Assim, a prosódia afeta a naturalidade e inteligibilidade e está relacionada com a presença e duração de pausas, o *pitch*, o valor da frequência fundamental, bem como duração e amplitude dos fones (SCHROETER, 2005; AZUIRSON, 2009; MAEDA, 1995).

Uma modelagem apropriada da prosódia é essencial para produzir falas com alto grau de naturalidade. Detalhes fonéticos, como nasalização de vogais, e melhorias nas fontes de excitação também são necessárias para obter uma "fala" natural. Muitas vezes, tais melhorias são feitas com base em experimentações de tentativa-e-erro (MAEDA, 1995).

Devido ao alto nível dos sistemas de processamento acústico existentes atualmente, a maior parte das pesquisas tem se voltado para modelagem linguística e prosódica (AZUIRSON, 2009).

A prosódia pode fornecer pistas sobre a estrutura sintática, resolvendo ambiguidades. Permite ainda a segmentação de enunciados longos em unidades menores. No caso da pronúncia, a prosódia é dependente do falante, incluindo gênero, tarefa específica, etc. (AZUIRSON, 2009; SCHROETER, 2005).

2.6.5 Entonação e Duração

A especificação automática da entonação a partir de um texto comum continua sendo um desafio para os sistemas de síntese de voz. Os sistemas de síntese de voz devem produzir uma entonação apropriada. Três parâmetros dinâmicos prosódicos, ou suprassegmentais, contribuem para a entonação: *pitch*, duração e amplitude. No nível segmental (fonema), a amplitude varia muito de acordo com a forma de articulação. Em geral, vogais são mais intensas que consoantes, e, obviamente, sílabas tônicas mais intensas que as átonas.

A duração está fortemente ligada com fonemas, sendo que as vogais são mais longas do que as consoantes, bem como sílabas tônicas em relação às átonas. A frequência fundamental ou *pitch* (F_0) é o parâmetro mais complexo, apresentando grandes mudanças em F_0 nas sílabas enfatizadas.

Especificar uma entonação natural é difícil. Há poucos indicadores confiáveis que ajudam a especificar efeitos entonacionais. Sistemas de síntese de voz inserem pausas após pontos finais, de interrogação, de exclamação, dois pontos e ponto-e-vírgula. Em muitas línguas, uma pausa entonacional ocorre após uma palavra de conteúdo, aquelas que contém informação, como substantivos, verbos, adjetivos e advérbios, e antes de palavras de função, como preposições, artigos, pronomes, etc. Em geral, locutores destacam a palavra final em uma sequência de palavras de entonação.

Tradicionalmente, a entonação é especificada por meio de regras baseadas em informações semânticas fornecidas por um bloco de processamento denominado NLP (*Natural Language Processor*), a ser detalhado mais adiante. entretanto, obter entonação direta e automaticamente por meio de treinamento é mais viável, não sendo necessário que especialistas interpretem dados manualmente (SHAUGHNESSY, 2003).

3. VISÃO GERAL E PROJETO DE UM SISTEMA DE SÍNTESE DE VOZ VIA *SOFTWARE*: ASPECTOS QUALITATIVOS E PROBLEMAS RELATIVOS

O presente Capítulo visa discutir o funcionamento geral de um sistema TTS, detalhando suas etapas de funcionamento, bem como as principais falhas realizadas por estes sistemas atualmente. Além disso, é apresentada a técnica de síntese de voz baseada em concatenação de unidades sonoras pré-gravadas, técnica esta utilizada nesta Dissertação, sendo discutidas de forma detalhada seu funcionamento, vantagens e desvantangens.

A voz é um dos melhores meios de interface, pois não requer treinamento, uma vez que é uma forma de comunicação natural (AZUIRSON, 2009).

A síntese de voz é a geração de um sinal de voz, podendo partir de uma transcrição fonética acompanhada da prosódia associada. Tal síntese é geralmente uma etapa de um sistema TTS, cuja entrada é um texto convencional. Assim, a síntese de voz é a produção artificial da voz humana, podendo ser implementada via *hardware* ou *software*. Muitos sistemas operacionais incorporaram sintetizadores de voz no início dos anos 90.

Sintetizadores de voz em geral requerem uma saída de áudio. A maioria dos *desktops* e *notebooks* vendidos atualmente dispõe de um suporte de áudio satisfatório. Evidentemente, quanto maior a qualidade da placa de som, melhor é o resultado da síntese, uma vez que, para que sejam executados de modo mais efetivo, alguns sintetizadores podem exigir configurações mais robustas, necessitando de mais memória ou maior poder de processamento.

O objetivo da síntese TTS é converter uma entrada de texto para uma saída de voz natural e inteligível para transmitir a informação da máquina para uma pessoa. A metodologia usada no TTS é explorar representações acústicas da fala para síntese, juntamente com análise do texto a fim de obter pronúncias corretas e prosódia de acordo com o contexto (SCHROETER, 2005).

Alguns sistemas TTS convertem textos convencionais diretamente para formas de onda, enquanto que outros se baseiam em representações simbólicas linguísticas, como transcrição fonética, para tal. Alguns sistemas se baseiam na concatenação de trechos de voz pré-gravados e armazenados em um banco de dados, enquanto que outros se baseiam na modelagem do trato vocal. Tais sistemas inicialmente realizam um processamento linguístico, produzindo a conversão "letra-para-som" a fim de gerar a

transcrição fonética correspondente ao texto de entrada, além das etapas de geração de prosódia e entonação. Tais etapas agem como um *front-end* geralmente (SHAUGHNESSY, 2003).

3.1 Aplicações das tecnologias de voz e suas vantagens

As tecnologias de voz estão se tornando cada vez mais importantes, tanto na computação pessoal como empresarial, e têm sido usadas para melhorar interfaces para os usuários já existentes e proverem suporte às novas formas de interações homem-máquina. Estas permitem o uso de computadores mantendo as mãos livres e/ou à distância. Entretanto, o reconhecimento e a síntese de voz podem melhorar a acessibilidade ao computador para usuários portadores de deficiência e podem reduzir os riscos de lesões por esforço repetitivo e outros problemas causados por outras interfaces atuais.

Tecnologias de voz podem aumentar as possibilidades com relação às tradicionais interfaces gráficas de usuário, permitindo comandos mais complexos do que "Sim, Não, Ok, Cancelar e Aplicar". Por exemplo: um comando "Usar tamanho 12, itálico, fonte Times New Roman" substitui diversos cliques em menus de seleções. Outras aplicações possíveis estão em ferramentas CAD que, enquanto se desenha, pode-se, simultaneamente mudar a cor e a espessura de uma linha, por exemplo, sem ter a necessidade de tirar o mouse dentro da área de desenho (SUN MICROSYSTEMS, 1998).

Sistemas de síntese de voz permitem, por exemplo, detectar erros gramaticais, ortográficos e estilísticos com maior facilidade, por ser mais fácil perceber tais erros ouvindo do que lendo ou informar ao usuário algum alerta sem abrir uma janela que interrompa visão do programa em execução: uma mensagem de alerta pode ser direcionada ao usuário sem que o mesmo desvie sua atenção para o objeto atual. Isto o deixa livre também para usar mãos e olhos em outras tarefas paralelas e conferindo maior agilidade na realização de tarefas, além da não obrigatoriedade do usuário de estar próximo ao dispositivo. Além disso, o uso de sistemas baseados em voz admite a interação através do telefone e garante acessibilidade aos deficientes visuais. Outras aplicações possíveis são: ensino de línguas estrangeiras, livros e brinquedos falantes (AZUIRSON, 2009).

Em salas de cirurgia, onde cirurgiões mantém suas mãos ocupadas e o eventual contato com teclados representam um risco à higiene, comandos de equipamentos médicos por voz podem facilitar o andamento de um procedimento cirúrgico.

Adicionalmente, os sistemas de voz têm sido amplamente utilizados em *call centers* de empresas, por oferecerem um meio de interação mais natural e substancialmente mais eficiente e rápido do que interfaces baseadas em digitação. Aplicações em empresas de telefonia com *hardware* dedicado capazes de suportar um grande número de conexões simultâneas por exemplo, usando cartões DSP, com capacidades para reconhecimento e síntese de voz podem ser substituídas em parte por tais aplicações.

Tecnologias de voz têm sido integradas em um grande número de sistemas embarcados de pequena escala como forma de reduzir mais ainda o tamanho, como PDAs (*Personal Digital Assistant*), brinquedos e controles de dispositivos em geral. Interações via voz podem ser uma alternativa mais atraente como interface em smartphones, ao invés da tecnologia *touch screen*. Além disso, podem permitir também que seja melhorada a experiência de navegação na Internet, possibilitando novas formas de navegação: o reconhecimento de voz pode ser usado para controlar navegadores, *applets*, preencher formulários, etc (SUN MICROSYSTEMS, 1998).

Ademais, o reconhecimento de voz pode ser usado para reforçar a segurança de um sistema, admitindo que alterações sejam feitas apenas após ser realizada a identificação do interlocutor por meio de voz (SUN MICROSYSTEMS, 1998).

A síntese de voz pode auxiliar na redução de espaço armazenado em disco em aplicações que façam uso de saídas de voz pré-gravadas em um fator de até 1000 vezes menor no tamanho de espaço de armazenamento exigido, além de remover as limitações impostas por sentenças pré-definidas (SUN MICROSYSTEMS, 1998).

Por fim, os comandos por voz são naturais e mais fáceis de lembrar que a localização de funções em menus e caixas de diálogo (SUN MICROSYSTEMS, 1998).

3.2 Visão geral de um sistema TTS

O diagrama de blocos de um sistema TTS convencional é mostrado na Figura 3.1. Um sistema TTS é composto por duas partes: um *front-end* e um *back-end*. O *front-end,* por vezes chamado de bloco de Processamento Linguístico-Prosódico, é composto por módulos NLP (*Natural Language Processing*) que correspondem aos blocos de Análise do Texto – e inclui as etapas de Pré-processamento e Análise Lnguística e Morfossintática, Análise Fonética e Análise Prosódica. Já o *back-end*, também chamado

de Bloco de Processamento Acústico, Motor de Síntese ou ainda Processador Digital de Sinais, é composto por módulos de processamento de voz, o motor de síntese, para a geração de voz sintetizada. O *back-end* possui um conjunto de filtros que recebem parâmetros amostrais de voz, juntamente com os rótulos de contexto prosódico para gerar a forma de onda de sinais de voz correspondente ao texto a partir dos fonemas e seus respectivos parâmetros prosódicos gerados pelo bloco de Processamento Linguístico-Prosódico. É possível perceber que o *front-end* é a parte mais próxima do texto de entrada, enquanto que o *back-end* é a parte do sistema mais próximo da saída falada. (COSTA e MONTE, 2012; AZUIRSON, 2009).

O *front end* é responsável por detectar e analisar a estrutura do texto de entrada e possui duas tarefas principais: a primeira é converter textos contendo símbolos, números e abreviações em sua forma por extenso em um processo chamado de normalização, pré-processamento ou ainda “tokenização”. A outra tarefa é a transcrição fonética. A transcrição fonética e a informação sobre a prosódia são utilizadas pelo *back-end*, ou sintetizador propriamente dito. Opcionalmente, o texto de entrada pode conter *tags* para o controle da prosódia e outras características.

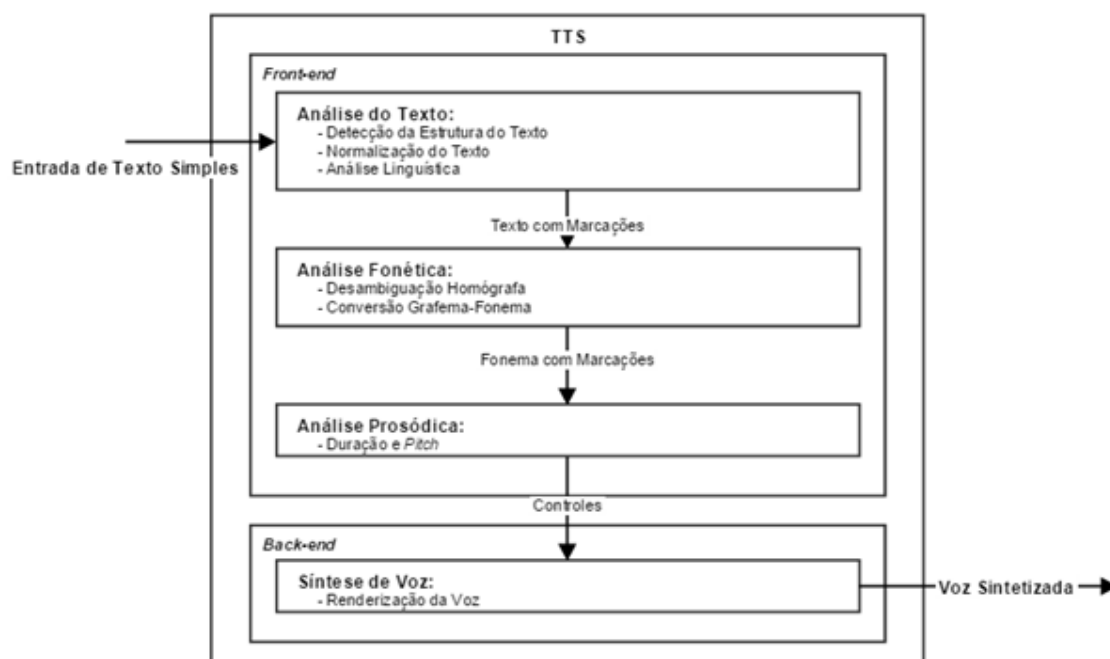



Figura 3.1: diagrama de blocos de um sintetizador de voz. Fonte: (SCHROETER, 2005 - Adaptado).

O *front-end* possui um conjunto de algoritmos que devem normalizar o texto, aplicar regras para conversão grafema-fonema, divisão silábica e marcação de sílaba

tônica. Estas informações são utilizadas para determinar as características prosódicas da fala. No HTS, *HMM-Based Speech Synthesis System*, por exemplo, as informações prosódicas são agrupadas em um arquivo chamado de rótulo de contexto e contém informações de diversos níveis, como fonemas, sílabas, palavras, frases, etc. (COSTA e MONTE, 2012).

A pontuação não é infalível. Em inglês, por exemplo, o ponto pode tanto representar separador decimal como fim de frase. Mapeamento de abreviações e siglas também podem ter resultados ambíguos. Por exemplo: DC pode significar Distrito de Columbia, mas também Corrente Contínua em inglês. Assim, a normalização de um texto e o módulo de normalização afetam fortemente a taxa de precisão de um sistema TTS, bem como a análise linguística, que é responsável por determinar sentido de palavras, ênfase, estilo de fala, emoções (SCHROETER, 2005).

A conversão grafema-para-fonema envolve a pronúncia de uma palavra, cujo mapeamento de sua ortografia para fonemas pode ser difícil por conta da dependência com o contexto em que se encontra. Em geral, tal problema é tratado com um treinamento de classificação e árvores de regressão, também chamadas de árvores de decisão, que capturam as probabilidades de conversões específicas, dado o contexto caso uma palavra seja uma homomorfa. Também são usadas regras letra-para-som. Nomes em geral costumam apresentar problemas também (SCHROETER, 2005).

Em resumo, a síntese de voz a partir de texto pode ser dividida em duas etapas: a primeira etapa corresponde à análise do texto e consiste em obter a representação fonética com base na ortografia do texto; e a etapa de síntese, que é a geração do sinal acústico associado à representação fonológica obtida no processo anterior. A etapa de análise do texto pode ser subdividida em subprocesos como o pré-processamento e o processamento prosódico (AZUIRSON, 2009).

O *front-end* tem a função de processar o texto e gerar como saída os fonemas correspondentes em conjunto com as suas respectivas informações a respeito da prosódia, duração e frequência. O *front-end* pode ser subdividido em outros módulos cuja saída de um serve de entrada para o bloco seguinte. Cada língua possui seu conjunto de fonemas básico, o que implica que a construção dos módulos que fazem parte do bloco de processamento linguístico e prosódico são dependentes da língua escolhida (AZUIRSON, 2009).

O processamento do texto é um processo mais próximo da modelagem da língua do que processamento de sinais propriamente dito. Como dito anteriormente, o

processamento de texto é feito por meio de um *front end*. A entrada de texto é transformada em representações que permitam acesso às unidades armazenadas em um banco de dados juntamente com informações adicionais de controle de entonação. Deve-se conhecer a sequência de fonemas, dífonos ou palavras, a serem pronunciados, quais sílabas são mais fortes, onde deve haver pausas entonacionais, etc. (SHAUGHNESSY, 2003).

Embora certos princípios do NLP possam parecer universais, línguas usam alfabetos diferentes, e cada língua tem um conjunto de fonemas. Especialistas em fonética estabeleceram um conjunto de fonemas universais, caracterizados pelo alfabeto fonético internacional, a partir do qual cada língua seleciona um subconjunto com pequenas diferenças articulatórias e acústicas (SHAUGHNESSY, 2003).

A técnica de síntese baseada em formantes (ver Apêndice B) pode ser facilmente modificada para uma nova língua, ajustando parâmetros fonéticos. Entretanto, sistemas baseados em concatenação ou LPC são menos flexíveis com relação ao ajuste de tais parâmetros (SHAUGHNESSY, 2003).

O passo inicial do NLP no TTS é a conversão de uma entrada de texto em um código que permita acesso ao banco de dados. No caso comum das unidades serem fonemas, isto é chamado de mapeamento texto-para-fonema ou ainda mapeamento letra-para-som e geralmente é feito por meio de “*Lookup Tables*”. Um dicionário também pode ser usado, incluindo todas as palavras, sua pronúncia - com marcação de sílaba tônica, categoria e informações sintáticas e semânticas. Alguns sistemas também possuem regras para prever a pronúncia: por exemplo, na língua inglesa, o som p é pronunciado como /p/, exceto quando sucedido por h. Em certas línguas, como o coreano e espanhol, tal mapeamento é simples, já que a há uma relação direta de um-para-um entre letras e fonemas. Outras línguas são descritas por um pequeno conjunto de regras, como o italiano e o alemão. Já outras são mais complexas, como o inglês e o chinês. Nos sistemas TTS mais sofisticados, os erros se limitam a nomes próprios e palavras estrangeiras não existentes no dicionário (SHAUGHNESSY, 2003).

3.2.1 módulo de processamento linguístico-prosódico

O objetivo da etapa de processamento linguístico-prosódico é obter uma sequência de unidades sonoras correspondentes ao texto de entrada em conjunto com os parâmetros referentes à prosódia.

Os sub-processos envolvidos nesta etapa estão mostrados na Figura 3.4, que são: Pré-Processamento, Análise Linguística e Morfossintática, Transcrição Fonética e Processamento Prosódico.

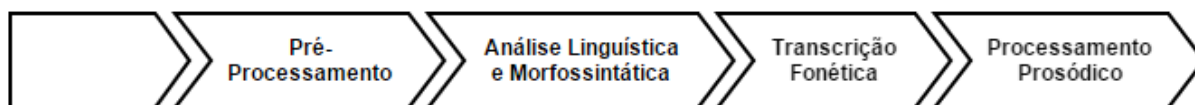



Figura 3.2: diagrama de blocos do bloco de processamento linguístico-prosódico. Fonte: (AZUIRSON, 2009).

3.2.1.1 Pré-Processamento

Um texto pode conter diversos símbolos e caracteres, sendo necessário converter tais símbolos em caracteres para que seja possível ser processado pelos módulos seguintes. Assim, caracteres especiais, como operadores aritméticos, sinais e outros símbolos como @, $, etc., siglas, abreviaturas e dígitos são pré-processados em uma etapa denominada normalização, na qual caracteres são substituídos por sua forma "por extenso". Em um texto genérico, a primeira tarefa consiste em tentar isolar as palavras para que possam ser corretamente processadas nas etapas posteriores, principalmente as seguidas por sinais de pontuação, como ".", ",", ";","?","!", aspas e parênteses. A diferença entre sentenças exclamativas, interrogativas e declarativas é fundamental para a entonação. Embora possa parecer uma tarefa simples de substituição, supressão e expansão de símbolos, pode ser extremamente complexa quando certas entradas são dependentes de contexto (AZUIRSON, 2009).

3.2.1.2. Análise Linguística e Morfossintática

A análise morfossintática é útil para resolver ambiguidades com relação à transcrição fonética. Por exemplo a palavra piloto pode ser pronunciada com som aberto ou fechado, dependendo de sua função morfológica, verbo ou substantivo, respectivamente. O mesmo acontece com as palavras "molho" e "seco". No português, muitas palavras são homógrafas mas não são homófonas, em outras palavras, apresentam exatamente a mesma grafia, porém são pronunciadas de forma diferente. Isto torna a etapa de Análise Linguística e Morfossintática muito importante, pois a estrutura prosódica de uma sentença está ligada à análise morfossintática (AZUIRSON, 2009).

Há casos em que a análise gramatical é insuficiente para resolver ambiguidades, neste caso, a análise semântica (significado das palavras) e pragmática (intenção do

falante), se faz necessária para a pronúncia correta. Entretanto, são poucos os sistemas que realizam análise semântica-pragmática.

A análise morfossintática é realizada por um *parser*. O *parser*, que é um analisador morfológico-sintático, é um módulo extremamente importante para assegurar a qualidade da síntese, uma vez que o mesmo é que permite a inferência dos constituintes prosódicos de uma sentença a partir de sua análise morfossintática (AZUIRSON, 2009).

3.2.1.3. Transcrição Fonética

A etapa de transcrição ortográfico-fonética consiste em representar uma sequência de palavras em uma sequência de símbolos fonéticos. Tal etapa é precedida por outras duas: a separação silábica e a determinação de sílabas tônicas. Esta última, além de ajudar a assegurar uma correta transcrição, é de grande utilidade para o processamento prosódico. O resultado da transcrição fonética é dependente da língua para o qual o sistema é projetado, pois o conjunto de fonemas e o mapeamento entre letras e fonemas varia bastante de uma língua para outra. Algumas línguas são bastante fonêmicas, ou seja, a escrita é muito próxima da expressão oral, como o russo, italiano e espanhol, quando comparadas a outras mais irregulares, como inglês e francês, e nesse caso, a dificuldade de realizar a transcrição é bem menor. A língua portuguesa é razoavelmente fonêmica, entretanto, ainda assim, a transcrição não é uma tarefa trivial (AZUIRSON, 2009).

A transcrição fonética é realizada com base em um conjunto de regras, sendo que algumas palavras fogem totalmente às regras de transcrição, algumas por serem estrangeiras. Este problema, entretanto, pode ser facilmente contornado por meio de um dicionário de exceções, contendo a palavra com sua respectiva transcrição fonética. A busca pela palavra no dicionário de exceções é a primeira tarefa realizada quando iniciado o módulo de transcrição fonética. Quando a palavra não é encontrada no dicionário, então se segue a divisão silábica, identificação da sílaba tônica e aplicação das regras de transcrição. Para o português, a aplicação direta das regras de transcrição é adequada para boa parte dos casos (AZUIRSON, 2009).

A determinação de sílabas tônicas é feita também por um conjunto de regras, sendo vital para o estudo da prosódia. Para palavras acentuadas, a identificação da sílaba tônica é evidente. Sendo que a dificuldade ocorre na identificação de oxítonas não acentuadas, uma vez que todas as proparoxítonas são acentuadas e a maioria das palavras não acentuadas no português são paroxítonas. Assim, algumas regras podem ser aplicadas para a identificação de oxítonas não acentuadas, como palavras terminadas

em "im" e "um"; palavras terminadas em "ar", "er" e "or", palavras terminadas em z antecedidas por vogais. Deve-se observar que essas regras não se aplicam a todos os casos, mas apresentam bom índice de precisão (AZUIRSON, 2009).

A transição entre palavras é um fator importante para assegurar a naturalidade da pronúncia. Um dos fenômenos a serem tratados no que diz respeito à co-articulação é o "sândhi externo" que ocorre na junção de palavras em que a última vogal da primeira palavra é igual à primeira vogal da segunda palavra. Neste caso, a coarticulação transforma a junção em uma vogal apenas. Outro fenômeno a ser observado é o fonema /s/, uma fricativa sonora surda, exceto quando seguida por palavra iniciando por vogal ou consoante sonora (AZUIRSON, 2009).

3.2.1.4. Processamento Prosódico

O processamento prosódico é a última etapa do bloco de processamento linguístico-prosódico, tendo como entrada a informação supra-segmental e segmental obtida pelas etapas anteriores, como as marcas prosódicas e transcrição fonética, a fim de traduzir variações de duração do segmento, ritmo, frequência fundamental, entonação, e inserção de pausas nas fronteiras prosódicas (AZUIRSON, 2009).

A prosódia é dividida em dois níveis: segmental e supra-segmental. O nível segmental se ocupa com a observação da variação dos parâmetros prosódicos, como a duração, frequência fundamental e amplitude, a nível de segmento, e supra-segmental. Esse nível foca na interação do segmento om seus vizinhos e a interferência dos vizinhos sobre o segmento observado. Já o nível supra-segmental se utiliza da estruturação da sentença a nível de sílabas, palavras, frases. Nesta etapa são usadas as seguintes informações dos módulos anteriores: determinação da sílaba tônica da palavra, da estrutura prosódia a partir da estrutura sintática, das pausas e da análise morfossintática das palavras (AZUIRSON, 2009).

O módulo de processamento prosódico é o último módulo antes do processamento acústico, sendo responsável por tratar a informação de módulos anteriores e fornecendo uma lista de fonemas em conjunto com parâmetros prosódicos (AZUIRSON, 2009).

Um destes parâmetros prosódicos é a duração, que mede a distância temporal do início ao término de um segmento fonético e que pode ser da ordem de dezenas a centenas de milissegundos (AZUIRSON, 2009).

Os fatores que determinam a duração de um fone são divididos em três grupos: os de natureza segmental, os de natureza coarticulatória e os de natureza supra-segmental. Os de natureza segmental são aqueles relacionados ao tipo de segmento. Já os de

natureza supra-segmental dependem do efeito prosódico desejado no instante em que o segmento ocorre. Os fatores de natureza coarticulatória serão discutidos no item 4.2.5 da presente Dissertação (AZUIRSON, 2009).

A duração de unidades fonéticas pode ser influenciada tanto pelo contexto fonético anterior como posterior, ou seja, a duração é calculada com base nos limites impostos pela concatenação com os segmentos vizinhos. Geralmente os falantes tendem a enfatizar palavras de conteúdo e colocar palavras funcionais em segundo plano, influenciando a duração dos fonemas (AZUIRSON, 2009).

A geração automática da duração de segmentos pode seguir dois modelos: estatísticos e baseados em regras (AZUIRSON, 2009). Os modelos estatísticos usam uma base de dados um dicionário de duração ou modelos baseados em *clustering* não hierárquico. Devido à coarticulação, por vezes é difícil saber onde começa e onde termina um segmento, sendo complexo marcar as fronteiras automaticamente (AZUIRSON, 2009).

O modelo de Klatt faz parte da classe dos modelos multiplicativos baseados em regras. Nos modelos multiplicativos, a duração de um fone é uma função de várias variáveis, cada uma responsável por influenciar a duração do fone. Baseado na língua inglesa, estabeleceu-se o seguinte conjunto de regras: cada segmento possui uma duração intrínseca, correspondente a um valor médio da distribuição dos valores que aquele segmento pode assumir e cada regra tenta prever a variação percentual a fim de efetuar um aumento ou diminuição na duração do segmento. Além disso, os segmentos não podem assumir valores menores que uma certa duração mínima (AZUIRSON, 2009).

A equação básica desse modelo pode ser expressa por (AZUIRSON, 2009):

$$D = D_{min} + \prod_{j=1}^{N} k_j (D_i - D_{min}), \qquad (5)$$

Em que D é a duração calculada para o segmento; D_i é a duração intrínseca para o segmento; D_{min} é a duração mínima para o segmento; k_j é um fator de ajuste da duração associada à regra j e n é número de regras aplicáveis ao contexto. Esta equação expressa a contribuição ponderada da diferença entre as durações intrínseca e mínima para a duração de cada segmento. Este cálculo é fundamental, pois uma duração correta faz com que o resultado se aproxime o máximo possível de um falante natural.

As regras se aplicam a fonemas, sílabas, palavras, constituintes prosódicos e sentenças, salientando que essas regras específicas para cada língua, sendo no caso do modelo de Klatt, a língua inglesa (AZUIRSON, 2009).

Nas regras definidas por Klatt, não foram determinados os valores dos parâmetros associados a cada regra. Tais valores de k são determinados por meio de ajustes empíricos sucessivos (AZUIRSON, 2009).

A determinação dos parâmetros prosódicos não é uma tarefa simples e não possui uma única solução possível, já que a prosódia é a marca da individualidade do falante. Isto explica o porquê do fato de uma sentença poder ser lida corretamente de várias formas diferentes (AZUIRSON, 2009).

Uma duração correta faz com que o resultado se aproxime o máximo possível de um falante natural (AZUIRSON, 2009).

3.2.2 módulo de processamento acústico

A produção da forma de onda, o passo final, utiliza as informações sobre a fonética e a prosódia para produzir a forma de onda do som de cada sentença. Há diversas formas nas quais o som pode ser produzido a partir dessas informações. A maioria dos sistemas atuais faz uso de uma das duas formas seguintes: concatenação de trechos de falas pré-gravadas, que pode consumir um grande espaço em disco, além de limitar as possibilidades de interação apenas ao que foi gravado anteriormente, ou usando algoritmos de processamento de sinais, por meio de modelos matemáticos baseados no conhecimento a respeito dos fonemas e métrica (SUN MICROSYSTEMS, 1998)

O módulo de processamento acústico é também chamado de processador digital de sinais ou motor de síntese. É a última etapa do processo TTS (AZUIRSON, 2009).

Todos os modelos de síntese de voz tem o mesmo objetivo, que é gerar sinal acústico correspondente à sequência de fonemas fornecida pelo módulo de transcrição ortográfico-fonético e aplicar os parâmetros prosódicos fornecidos pelo módulo de processamento prosódico (AZUIRSON, 2009).

Sintetizadores de voz podem apresentar erros em qualquer uma das etapas de síntese descritas anteriormente. O sistema auditivo humano é sensível a esses erros, de tal forma que os desenvolvedores devem minimizar esses erros e melhorar a qualidade do som resultante na saída.

3.3 Síntese de voz baseada em concatenação

A síntese de voz baseada em concatenação é gerada a partir da concatenação de segmentos de voz armazenados em um banco de dados de referência. Geralmente é a técnica que produz resultado mais natural (SHAUGHNESSY, 2003).

A principal limitação para a síntese de formantes e síntese articulatória é gerar voz a partir de representação paramétrica, principalmente no que diz respeito a encontrar trais parâmetros, a partir do resultado do processo de análise do texto. A síntese concatenativa adota uma abordagem orientada a dados.

Nos anos de 1970 e 1980, computadores eram capazes de realizar boas sínteses, mas as limitações de memória permitiam que apenas pequenas unidades sonoras fossem armazenadas e concatenadas. Assim, se até recentemente os métodos espectrais eram as técnicas dominantes, a simplicidade de se concatenar unidades de formas de onda aliado à capacidade de armazenamento dos computadores fez com que tal técnica voltasse a receber atenção. As primeiras tentativas de síntese baseadas em "colagens" não apresentaram resultados satisfatórios. Atualmente, a maioria dos sistemas TTS em desenvolvimento são baseados em metodologias de concatenação de formas de onda. A técnica PSOLA, por exemplo, aumentou significativamente a qualidade de um sistema TTS, sendo, atualmente, a qualidade em geral comparável aos demais sistemas mais avançados baseados em regras disponíveis no mercado (SHAUGHNESSY, 2003; MAEDA, 1995).

A síntese concatenativa é mais simples que a síntese baseada em regras e parâmetros para simular fonemas e suas transições, uma vez que não é necessário determinar regras para a síntese, baseando-se apenas na justaposição de segmentos de voz natural pré-gravados, o que elimina a necessidade de ter conhecimentos detalhados sobre a fala (AZUIRSON, 2009; MAEDA, 1995)

Teoricamente, a síntese concatenativa deveria apresentar qualidade inferior em decorrência da descontinuidade – resultante da destruição da coerência física do sinal em cada ponto de concatenação, o que pode ser contornado ao se aumentar o tamanho das unidades sonoras. Reduzir as descontinuidades na transição espectral e o uso de algoritmos de concatenação capazes de modificar a envoltória espectral do sinal pode suavizar as descontinuidades (AZUIRSON, 2009).

Curiosamente, embora haja divergências sobre qual a abordagem mais promissora atualmente, a abordagem concatenativa produz resultados de síntese superiores, pois usa gravações de vozes humanas. Esta usa segmentos reais curtos de vozes gravadas que são

cortadas durante gravações e armazenadas em um inventário, um banco de dados de voz, tanto como formas de onda ou codificados por meio de um codificador adequado (SCHROETER, 2005).

A Figura 3.3 mostra o diagrama de blocos de um sistema baseado em síntese concatenativa genérico.

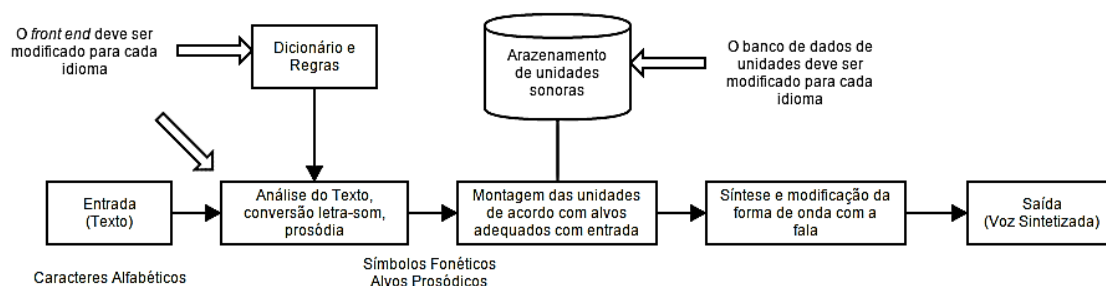



Figura 3.3: diagrama de blocos da síntese concatenativa. Fonte: (SCHROETER, 2005 - Traduzido).

O *front-end* de um sistema concatenativo deve converter uma entrada de texto em cadeia de caracteres (*string*) de símbolos fonéticos e informações de prosódia, como a frequência fundamental, duração e amplitude. O *front-end* emprega um conjunto de regras e/ou um dicionário de pronúncia. Juntamente com uma *string* de símbolos fonéticos, produz valores para frequência fundamental (*pitch*), duração de fonemas e amplitudes. A etapa seguinte monta as unidades de acordo com uma lista de alvos fornecidos pelo *front-end*. Tais unidades são selecionadas dentro do inventário de unidades sonoras disponíveis (SCHROETER, 2005).

3.3.1 Desvantagens

Se por um lado, a síntese concatenativa se destaca por gerar resultados com alta qualidade e com baixo custo computacional, por outro, sua desvantagem reside no fato de, por utilizar pedaços de fala, pode gerar descontinuidade espectral, resultando em voz metálica. Entretanto, tal efeito pode ser minimizado com a seleção e construção mais rigorosa do inventário (AZUIRSON, 2009).

Outra desvantagem reside no fato da falta de flexibilidade. Os segmentos de forma de onda existentes no inventário são construídos a partir de um falante em particular. Caso se deseje uma nova voz, deve-se construir um novo inventário de formas de onda com um novo falante (MAEDA, 1995).

Além disso, devido ao fato do banco de dados ser de tamanho finito, é impossível alcançar todas as possíveis variantes existentes na fala natural. Para que seja possível,

tem-se de lançar mão de técnicas que modifiquem a voz gravada em termos de dinâmica articulatória, timbre, ritmo e entonação. Técnicas no domínio do tempo são utilizadas para modificar o *pitch* e a duração, mas poucas técnicas concatenativas realizam alguma modificação espectral nas unidades sonoras. Uma destas poucas modificações consiste na normalização das diferenças acústicas existentes entre diferentes sessões de gravação, além da suavização de erros existentes durante a concatenação, como a técnica LPC excitada residual, que usa filtragem inversa e permite a perfeita reconstrução de sinal. Entretanto, esta técnica também apresenta suas falhas. No trabalho (WOUTERS et. al. 2000) é possível encontrar um estudo acerca de uma das estratégias para lidar com esta situação.

Outro ponto negativo é o fato de ser mais difícil modificar a prosódia, além de apresentar problemas de descontinuidade nas extremidades das unidades, podendo gerar resultados pouco naturais. Suavizar formas de onda é geralmente mais simples que uma suavização espectral, entretanto, o resultado soa mais descontínuo (TABET, 2011; SHAUGHNESSY, 2003).

Em resumo: embora extremamente eficiente e amplamente usado, é possível ouvir os pontos de concatenação, uma vez que o algoritmo não apresenta uma forma de suavizar as transições, que ocorrem abruptamente, pois as muitas mudanças de tom acompanham concatenações (SCHROETER, 2005; SHAUGHNESSY, 2003).

3.3.2 A escolha das unidades e dífonos

Como falado anteriormente, a síntese concatenativa explora vozes gravadas que compõem um inventário (SCHROETER, 2005).

Ao se concatenar unidades sonoras, a sucessão de tais unidades deve ser contínua. Uma vez que as unidades sonoras ao longo do treinamento são extraídas a partir de sinais de voz diferentes, a continuidade - tanto em amplitude como frequência espectral, não é garantida nos contornos durante a concatenação. As unidades são frequentemente escolhidas tomando a amplitude espectral como critério, reduzindo os problemas de continuidade espectral neste domínio. Entretanto, a fase espectral é mais complicada. Unidades consistem frequentemente de períodos completos de *pitch* (SHAUGHNESSY, 2003).

Para concatenação, podem-se usar fonemas, dois fonemas, sílabas, frases, palavras, frases, etc. Alguns trabalhos tem procurado usar unidades de tamanho variável. O tamanho das unidades a serem guardadas no banco é importante para a qualidade do resultado: quanto maior o tamanho de uma unidade, menor o número de junções no

resultado, logo, melhor a qualidade, resultante da menor geração de pontos de concatenação. O problema dos pontos de concatenação reside no fato de que é possível que as unidades provenham de contextos fonéticos diferentes e quando as unidades provêm de diversas fontes ou sessões de gravação, as unidades apresentam alto grau de descontinuidade nos contornos (TABET, 2011; AZUIRSON, 2009; SHAUGHNESSY, 2003).

Se por um lado, aumentar o tamanho dos seguimentos a serem concatenados pode melhorar a qualidade da voz sintetizada, por outro, o número de segmentos necessários aumenta dramaticamente, fazendo o espaço exigido para armazenamento também crescer. Além disso, o número de contextos aumentados dificulta a construção do banco de dados, o que significa que é necessário um grande conjunto de unidades a fim de se adequar a qualquer tipo de aplicação (AZUIRSON, 2009; SCHROETER, 2005).

Em outras palavras, o comprimento da unidade afeta a qualidade da síntese: quanto maior a unidade, maior a qualidade (naturalidade), pois são necessários menos pontos de concatenação, entretanto, o número de unidades armazenadas no banco de dados se torna muito numeroso. À medida que o tamanho das unidades cresce, o espaço para armazenamento cresce exponencialmente. Tornando-o até mesmo inviável. Usar pequenas unidades requer menos espaço para armazenamento, mas geralmente provê saídas menos naturais que quando usadas unidades maiores (KANG et. Al. 2009; SHAUGHNESSY, 2003).

No que diz respeito à construção do banco de dados com unidades menores, a coleta de unidades e as técnicas de rotulação se tornam mais complexas (TABET, 2011).

Do ponto de vista da flexibilidade dos sistemas, se as unidades são sentenças completas, a qualidade soa natural. Entretanto, tais sistemas são inflexíveis. Assim, para sistemas de vocabulário ilimitado, os bancos de dados armazenam um grande número de unidades, geralmente fonemas, dífonos e outras unidades.

O emprego de palavras como unidades básicas é inviável quando se deseja construir um sintetizador genérico. A utilização de palavras como unidades básicas geralmente se dá em contextos de vocabulário limitado (AZUIRSON, 2009).

Armazenar todas as palavras é impraticável também devido à enorme demanda exigida para o locutor que deverá ler centenas de milhares de palavras de uma forma consistente. E mesmo que tal tarefa fosse realizada em múltiplas sessões ao longo de

semanas, a falta de coarticulação e os contornos das palavras resultaria em uma fala pouco natural. (SCHROETER, 2005).

Sílabas têm sido sugeridas como unidades, porém as desvantagens superam as eventuais vantagens. Em inglês, por exemplo, são necessários em torno de 10000 sílabas para que seja possível formar todas as palavras. Adotando-se 10 frames por sílaba, o espaço para armazenamentos cresce substancialmente (SHAUGHNESSY, 2003).

Na língua chinesa, por exemplo, em síntese baseadas em sílabas, o fenômeno de coarticulação aparece apenas quando uma sílaba termina em vogal e a seguinte inicia em vogal ou som aspirado (KANG et. Al. 2009).

Utilizar fonemas como unidades pode fornecer grande flexibilidade e economia, entretanto pode apresentar problemas de coarticulação, tornando a inteligibilidade muito baixa. Sons das línguas podem ser descritos por apenas aproximadamente 100 fonemas e 30 diacríticos. A língua inglesa, por exemplo, possui 40 fonemas (SCHROETER, 2005; SHAUGHNESSY, 2003; AZUIRSON, 2009; MAEDA, 1995).

Entretanto, descrever sentenças por fonemas é muito pouco prático. Além disso, todos os esforços para concatenar segmentos do tamanho de um fonema têm mostrado resultados insatisfatórios. Isto resulta do fato da manifestação acústica dos fonemas depender fortemente do contexto segmental. É importante frisar também que a intensidade deve ser ajustada quando se concatena fonemas (MAEDA, 1995; SHAUGHNESSY, 2003).

Outra desvantagem é que, na concatenação por fonemas, ao se observar o espectro da voz, percebe-se que a quase totalidade da energia de uma palavra se encontra nas vogais, dificultando a inteligibilidade das consoantes quando armazenadas em separado. A síntese por dífonos contorna este problema, além de evitar problemas causados pela variabilidade de contexto (MACHADO, 1997).

Dífono é uma unidade sonora que começa na metade de um fonema e se estende até a metade do fonema seguinte. A metade de um fonema tende a ser a região mais estável acusticamente. Assim, o dífono representa a transição acústica da metade estável de um fonema. Uma vez que os limites de um dífono estão na metade dos fonemas, seu comprimento é o mesmo de um fonema, e não o dobro como inicialmente se possa esperar (TABET, 2011; SCHROETER, 2005; TALAFOVÁ et. al., 2007).

A ideia básica consiste em concatenar partes apenas estáveis do som, fazendo uso da região de transição entre as mesmas, como o meio de uma vogal e armazenar essas

informações em um inventário. Exemplo, "Paris" é resultado da concatenação de seis dífonos: <#p><pa><ar><ri><is><s#>, em que # denota o silêncio existente no contorno entre as palavras.

A curva de transição entre dois fonemas é mostrado na Figura 3.4, em que é possível identificar as regiões de transição, os núcleos dos fonemas e as descontinuidades existentes.

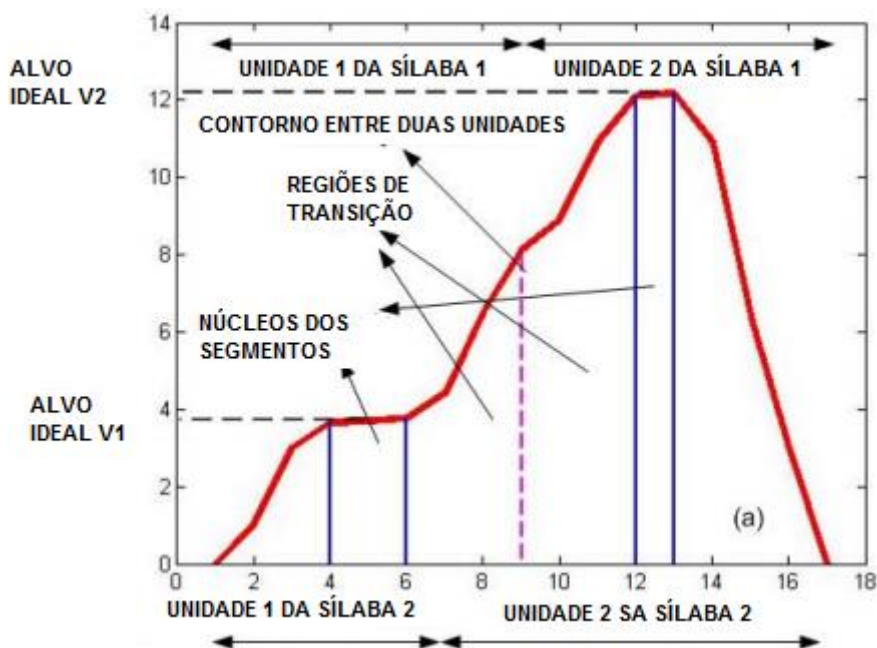



Figura 3.4: transição entre unidades sonoras. Fonte: (PHUNG et. al. - Traduzido).

Em termos de desempenho, isto faz com que dífonos apresentem melhor resultado na transição entre sons, uma vez que seus limites estão na metade dos fonemas e apresentam uma curva característica mais estável ao longo do tempo. Logo, os dífonos são vantajosos por conterem dentro delas mesmas o modelo de coarticulação (transição). Por uma questão de flexibilidade e economia, os dífonos são a unidade sonora mais usada na síntese concatenativa (TALAFOVÁ et. al., 2007; TABET, 2011).

Dífonos são úteis em síntese de voz por apresentarem resultados mais naturais do que simplesmente combinando fonemas por conta das variações de pronúncia destes últimos. Outra vantagem está no fato dos dífonos preservarem a informação da transição entre os fonemas, sendo guardados em um banco de unidades. Entretanto, ainda se faz necessário o uso de técnicas para suavizar a concatenação entre as unidades. Devido ao fato da síntese de dífono preservar os detalhes acústicos da fala natural, a síntese

baseada em dífonos é geralmente bastante inteligível (AZUIRSON, 2009; SCHROETER, 2005).

Se por um lado, os dífonos apresentam mesmo tamanho de um fonema, sejam N o número de fonemas de uma língua, teoricamente são necessários N^2 dífonos para construir um banco de dados de dífonos. Entretanto, todas as línguas apresentam restrições sobre quais sons são possíveis ou não de acontecer, o que torna o número de dífonos em cada língua muito menor que N^2, como no caso do espanhol, que apresenta 800 dífonos aproximadamente, enquanto que o alemão apresenta em torno de 2500. Desta forma, um banco de dados de dífonos é bastante viável, sendo necessário apenas alguns milhares janelas de dados espectrais (TALAFOVÁ et. al., 2007).

A lista completa de dífonos é denominada de inventário de dífonos. Para construir um inventário de dífonos deve-se gravar todos os fonemas em todos os contextos possíveis, e então tais dífonos são rotulados e segmentados.

Uma síntese baseada em dífonos usa um banco de dados mínimo contendo todos os dífonos existentes em uma língua. A prosódia é determinada por meio de técnicas de processamento digital de sinais como codificação preditiva linear, PSOLA, MBROLA ou técnicas mais recentes como a modificação do *pitch* por meio da transformada cosseno discreta. A síntese de dífono apresenta as mesmas deficiências presentes nas técnicas concatenativas, resultando em vozes pouco naturais e robóticas. Na síntese baseada em dífonos, apenas um exemplar de cada dífono é armazenado no banco de dados.

Uma vez construído o inventário, o *pitch* e a duração de cada dífono deve ser modificado a fim de atender à prosódia especificada (TABET, 2011).

Em certos casos, é difícil determinar a parte estável em um fonema. Nesses casos, podem-se usar dífonos silábicos ou trífonos. Neste último caso, Paris seria resultado da seguinte concatenação: <#pa><ari><is#>.

Por exemplo, para a língua inglesa, o número de fonemas, dífonos e trífono é, respectivamente 40, 1600, 64000 aproximadamente (MAEDA, 1995).

A desvantagem da síntese por dífonos é que a coarticulação é apenas dada apenas pelos fonemas precedentes e seguintes. Nesse caso, as semi-sílabas são uma alternativa interessante de serem consideradas. A semi-sílaba, como o próprio nome já sugere, é a metade de uma sílaba, compreendendo a parte inicial da primeira metade no do núcleo da sílaba ou a porção final da segunda metade do núcleo da sílaba. Devido ao fato de semi-sílabas serem unidades sonoras mais longas que dífonos, e permitem melhor

efeitos de coarticulação quando comparadas com dífonos, elas apresentam menos problemas de concatenação (SCHROETER,2005).

Uma generalização dos dífonos são os polifones, que são unidades que vão desde a região estável de um primeiro fonema até a região estável de um outro fonema, juntamente com a realização acústica completa de fonemas intermediários (AZUIRSON, 2009).

3.3.3 PSOLA / TD-PSOLA

PSOLA (*Pitch Synchronous Overlap and Add*) é uma técnica de processamento digital de sinais usada para síntese de voz criado em 1986 utilizado para modificar o *pitch* e a duração de um sinal de voz, com baixa complexidade computacional e no domínio do tempo.

PSOLA funciona dividindo a forma de onda em segmentos sobrepostos. Para modificar o *pitch*, os segmentos são afastados para diminuir o *pitch* ou aproximados para aumentar o *pitch*. Para modificar a duração do sinal, os segmentos são repetidos diversas vezes para aumentar a duração ou são eliminados para diminuir a duração, portanto, trata-se de uma técnica para escalonamento de tempo (duração) e escalonamento de *pitch* (MAEDA, 1995).

Os segmentos são combinados usando a técnica "*overlap add*". PSOLA pode ser usado para modificar a prosódia do sinal de voz (MAEDA, 1995), sendo a técnica não-paramétrica mais conhecida para este fim.

O método se baseia no uso de pontos de excitação de voz encontrados como método para análise de instantes de tempo para controle prosódico (MAEDA, 1995).

O PSOLA modifica o *pitch* conforme é mostrado nas Figuras 3.5 e 3.6: Uma janela pequena de tempo é aplicada à forma de onda original a cada análise de instante de tempo (períodos de *pitch*). A síntese é então feita simplesmente colocando essas janelas sobre essas formas de onda (*wavelets*). A modificação da duração é feita por meio da duplicação de uma ou mais *wavelets* para aumentar a duração ou eliminando (descartando) uma ou mais *wavelets* para encurtar a duração. Deve-se observar que este método funciona apenas no domínio do tempo. O intervalo de modificação de *pitch* varia de 0,5 a 2, suficiente para aplicações TTS, uma vez que o alcance do *pitch* é um falante é inferior a uma oitava (MAEDA, 1995).

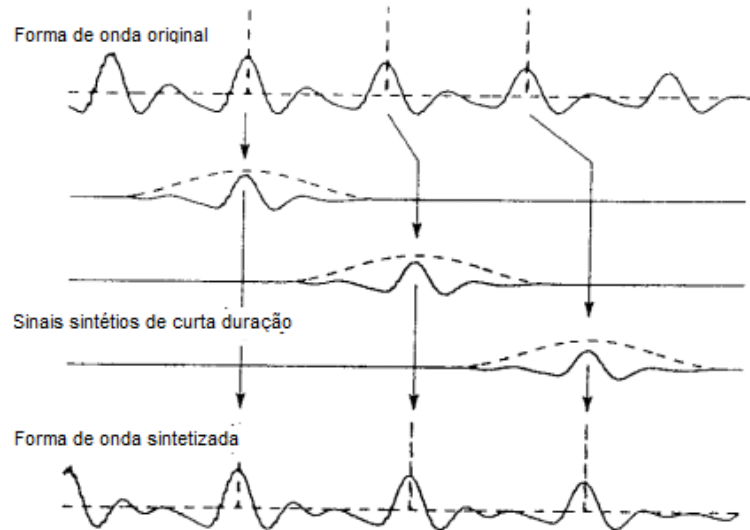



Figura 3.5: escalonamento de *pitch* e duração pelo PSOLA. Fonte: (MAEDA, 1995 - Traduzido).

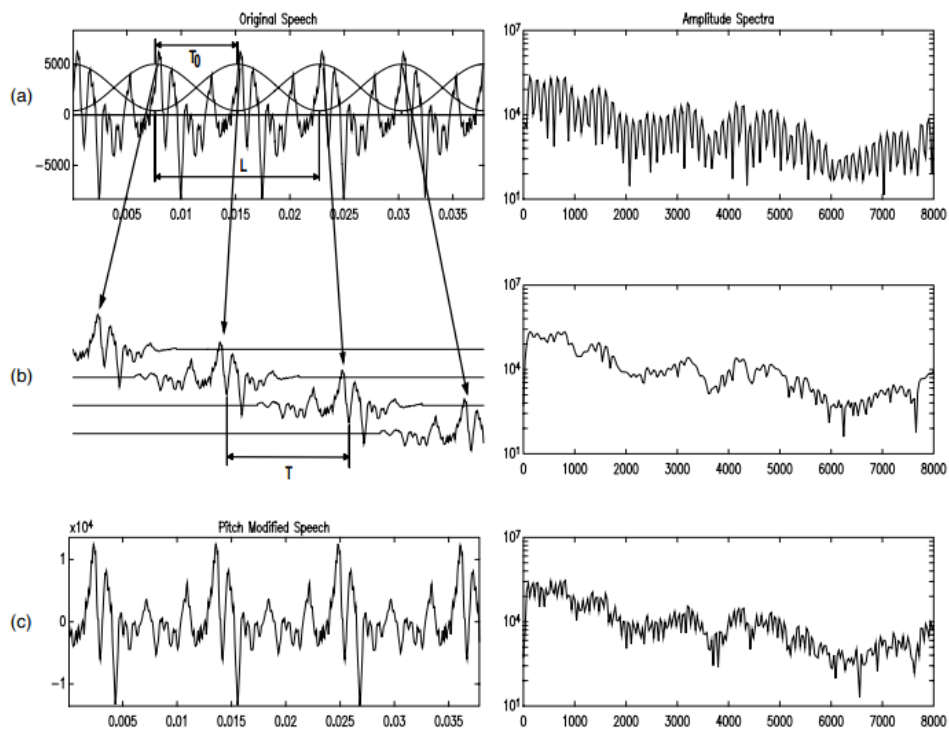



Figura 3.6: esquerda: domínio do tempo, direita: espectro. Fonte: (SCHROETER, 2005).

Uma vez que o método PSOLA processa o sinal no domínio do tempo, o algoritmo melhora o erro de modelagem da produção da voz e a distorção do espectro. Além disso, é mais adequado para o controle de prosódia em tempo real por apresentar menor tempo de processamento. Entretanto, esta técnica causa degradação da qualidade quando se combinam dados de sílabas extraídos de uma palavra diferente. Ademais,

causa um desequilíbrio de energia devido à aplicação de uma janela simétrica em um sinal de voz assimétrico (JUNG, 2001).

Se o sinal de voz é vozeado, o sinal de voz é feito por meio de um trem de sinais curtos após a multiplicação de uma função janela por um período de *pitch* decomposto. Se o som é não vozeado, este é analisado com 10ms. Pode-se usar as janelas de Hanning e de Hamming – Equaçoes 6 e 7, respectivamente (JUNG, 2001).

$$W(n) = \frac{1}{2}\left\{1 - \cos\left(\frac{2 \cdot \pi \cdot n}{N - 1}\right)\right\}, 0 \leq n \leq N - 1 \quad (6)$$

$$W(n) = 0{,}54 - 0{,}46\left\{\cos\left(\frac{2 \cdot \pi \cdot n}{N - 1}\right)\right\}, 0 \leq n \leq N - 1 \quad (7)$$

O período de *pitch* decomposto é obtido pela multiplicação do sinal de voz pela função janela com propriedade simétrica mostrada na Equação 8 (JUNG, 2001):

$$S_{análise}(n) = W_{análise}(m - n)S(n) \quad (8)$$

Em que $S_{análise}(n)$ é o pequeno sinal do período de *pitch*; $W_{análise}(n)$ é uma função Janela; m é o m-ésimo *pitch* e é $S(n)$: sinal de voz original.

A fim de modificar o *pitch*, o período do *pitch* é rearranjado por meio da alteração do seu período (JUNG, 2001):

$$S_{síntese}(n) = S_{análise}(n - m_a), \quad (9)$$

em que $S_{síntese}(n)$ é o período do *pitch* do sinal amostrado, m_a é o período do *pitch* a ser alterado.

A modificação do *pitch* é necessária para o controle da prosódia e para fornecer uma variedade de vozes e garantir uma maior qualidade na saída (JUNG, 2001).

Em geral, a modificação do *pitch* no domínio da frequência degrada a qualidade devido ao fato de, apesar de ter uma pequena distorção no espectro, é difícil de manter a fase. Por outro lado, a modificação do *pitch* no domínio da frequência pode manter a fase mas causar uma grande distorção no espectro devido à mudança na estrutura dos formantes (JUNG, 2001).

Além disso, a técnica PSOLA convencional cria trem de pequenos trechos de um sinal de voz original por meio da multiplicação do período de *pitch* decomposto com a função janela após decompor o sinal de voz. A fala é sintetizada a partir de uma unidade controlada após o controle da prosódia. Entretanto, a técnica PSOLA convencional adapta uma janela simétrica mesmo em um sinal assimétrico, causando desequilíbrio de energia, em outras palavras, ao aplicar uma função janela simétrica para uma forma de

onda assimétrica ocasiona o fenômeno de desbalanceamento de energia, sendo necessária uma normalização para manter a energia constante (JUNG, 2001).

O espectro do trato vocal representa a frequência de ressonância e é o mesmo espectro formante (JUNG, 2001).

O sinal de voz é sintetizado a partir da convolução da característica do trato vocal a baixas frequências com a excitação a altas frequências. O *pitch* pode ser alterado por meio da modificação da excitação característica (JUNG, 2001).

Em (JUNG, 2001) é apresentada uma solução para o problema do desbalanceamento de energia causada pela modificação do *pitch* no PSOLA, iniciando com a conversão tempo-frequência de uma forma de onda assimétrica para uma forma de onda simétrica (JUNG, 2001).

TD-PSOLA (*Time Domain Pitch-Synchronous Overlap Add*) consiste em cortar exatamente dois períodos de pitch de um sinal de voz, realizando janelamento a cada segmento com uma janela de Hanning centrada no ponto de fechamento glotal (máxima excitação) (SCHROETER, 2005).

O TD-PSOLA realiza uma sincronização do *pitch* automaticamente: períodos do *pitch* são extraídos, sobrepostos e somados a diferentes taxas a fim de produzir a saída. Ou seja, o sinal original $s(n)$ é decomposto e uma sequência de curtos sinais sobrepostos $s_m(n)$ usando uma janela de Hanning $h_m(n)$, centrada na origem $n=0$ (KOBAYASHI et. al., 1998).

Uma variante do TD-PSOLA pode apresentar um filtro LPC, que permite suavizar o envelope espectral nos pontos de concatenação (SCHROETER, 2005). Há outras variantes que usam modificações do modelo baseado em LPC, ou ainda modelos híbridos como o *Harmonic-plus-Noise Model* (HNM), mostrado na Figura 3.7. Este último faz uso do fato do espectro da voz em geral ser composto de duas partes distintas: a harmônica (periódica) - cuja maior parte está nas baixas frequências e é altamente relevante para um locutor específico, e a parte estocástica (ruído), existente nas frequências mais altas. Assim, dois sintetizadores separados são podem ser usados: um sintetizador harmônico e um sintetizador baseado em LPC com uma excitação estocástica (ruído) filtrada por um filtro passa alta. O sintetizador harmônico é controlado por parâmetros como frequência fundamental ω_0, amplitudes α_0 e fases φ_0 para a i-ésima harmônica e os parâmetros para um filtro variante no tempo opcional com resposta ao impulso $h_p(n,m)$, sintetizando a forma de onda $s_p(n,m)$. O sintetizador estocástico consiste de um filtro variante no tempo com resposta ao impulso

$h_\Gamma(n,m)$, um sinal de excitação $e_\Gamma(n)$, criando uma forma de onda $s_\Gamma(n)$. Ambos os componentes são adicionados ao sinal de banda completa $s(n)$. O HNM e abordagens similares permitem uma suavização da excitação nos pontos de concatenação. Uma desvantagem dos sintetizadores híbridos como o HNM reside na elevada complexidade computacional (SCHROETER, 2005).

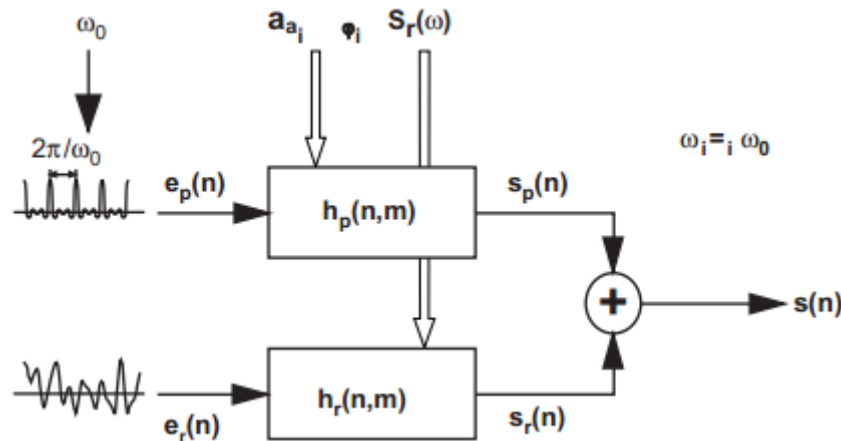



Figura 3.7: síntese HNM.

3.3.4 A criação do banco de dados de voz

Antes de iniciar a síntese de voz baseada em dífonos, deve-se criar um banco de dados de dífonos. O banco de dados consiste em gravações reais que são quebradas em partes menores, os dífonos. Além das vozes, que podem ser gravadas em um arquivo .wav, deve-se incluir um índice listando os dífonos e seus limites. Caso um determinado dífono não tenha sido incluído no banco de dados, pode-se fazer uso dos fonemas com o qual o mesmo é composto (TALAFOVÁ et. al., 2007).

Um banco de dados típico, cobrindo todas as unidades de dífonos possíveis para um conjunto de sentenças mínima, deve conter ao menos 30 minutos de vozes faladas, dado que tais unidades devem ser modificadas por meio de processamento de sinal a fim de se adequarem de acordo o requerido pelo *front-end* e apresentar pontos de concatenação suaves. Sistemas de alta qualidade podem apresentar um banco de dados com horas de gravações, não necessitando de modificações por já conterem em seu inventário um fragmento adequado (SCHROETER, 2005).

Experts são responsáveis por rotular espectrogramas e formas de onda, baseados em habilidades de escuta sofisticadas a fim de produzir anotações que incluem: marcações temporais, fim de palavras, representações para sílabas tônicas, melodias, fonemas,

pausas, etc. Experimentos mostram que tais profissionais precisam de aproximadamente de 100 a 250 segundos de tempo de trabalho para rotular um segundo de fala. Entretanto, a realização de tal tarefa manual é impraticável para grandes bancos de dados, que podem conter até dúzias de horas de gravações, sendo necessário fazer uso de sistemas automatizados, alguns inclusive baseados em sistemas de reconhecimento de voz. A vantagem é que tais sistemas de reconhecimento tem atingido alto grau de confiabilidade a ponto de apresentarem resultados até mesmo superiores que aquele feito por profissionais especialistas. As ferramentas de rotulação automática podem ser classificadas e duas categorias: ferramentas de rotulação fonética automática, responsáveis por rotular fonemas de forma adequada, e ferramentas de rotulação prosódica automática, responsáveis por rotular tons e tonicidade bem como pausas de forma adequada. É importante que tanto o sistema TTS baseado banco de dados a ser rotulado como a ferramenta de rotulação sigam uma convenção comum (SCHROETER, 2005).

O sinal de voz é armazenado em um formato comprimido de tal forma que o banco de dados de voz pode ser usado em sistemas com limitações de memória, de preferência com codificadores e decodificadores de baixo custo computacional, transparentes ao usuário e que permitam acesso aleatório (SCHROETER, 2005).

Deve-se tomar alguns cuidados ao se gravar vozes para o banco de dados: qualidade da gravação, escolha adequada da voz, definição e marcação adequada dos limites dos dífonos e equalização apropriada (TALAFOVÁ et. al., 2007).

Seleção adequada de locutor, com fala correta e consistente, e equipamento de gravação, em um ambiente livre de ruídos e reflexões acústicas, garante um banco de dados com qualidade boa o suficiente para realizar sínteses inteligíveis (SCHROETER, 2005).

É difícil para um locutor manter um estilo de fala uniforme por mais que algumas centenas de unidades, geralmente seleciona-se apenas algumas unidades desejáveis ao longo de diversas sessões de gravação. Assim, para tornar isto possível, algumas soluções apresentam seleção automática de unidades. Outros trabalhos apresentam soluções de como ajustar as junções por meio de otimização da "distância de similaridade", a fim de reduzir as descontinuidades. Tais distâncias devem contemplar não apenas o envelope espectral, mas também a continuidade de fase.

3.3.4.1 Compressão do banco de dados

Uma vez que o modelo TD-PSOLA não requer nenhum estágio de estimação de parâmetros, exceto marcação de *pitch*, este não está ligado a nenhum algoritmo de redução de dados, ou, em outras palavras, o TD-PSOLA pode ser, *a priori*, associado a qualquer técnica de codificação e compressão de voz. Deve-se levar em conta, entretanto, o equilíbrio entre a redução de espaço utilizado realizado por determinada técnica de compressão e a distorção que o mesmo insere no processo de síntese. Este equilíbrio pode ser medido pelo custo computacional. Tal equilíbrio reduz significativamente o número de algoritmos de compressão aplicáveis.

Técnicas de codificação de formas de onda tipicamente requerem pouco poder computacional, entretanto, oferecem baixas taxas de compressão. O codificador DPCM tem se mostrado o mais adequado para trabalhar com o TD-PSOLA (DUTOIT, 1997).

3.3.5 Problemas de Coarticulação

Coarticulação é um fenômeno fonológico que ocorre em todas as línguas sempre que há uma sequência de sons não separadas por pausas, referidas como a sobreposição de gestos articulatórios, sendo um fenômeno da ocorrência de dois ou mais características fonéticas afetadas antes ou depois dos fonemas durante a articulação. Portanto, trata-se de um processo relativamente comum durante a fala. Coarticulação é um problema para unidades sonoras de qualquer tamanho, entretanto, ao se concatenar unidades como palavras ou frases, há muito menos junções (SHAUGHNESSY, 2003).
Nas técnicas de concatenação de formas de onda atuais, para algumas línguas, como o Chinês, o tal efeito não é considerado, o que resulta em uma síntese da coarticulação ineficiente na junção das sílabas, reduzindo a naturalidade da fala sintetizada.

Por meio de um espectrograma é possível ver o deslocamento suave da energia durante a coarticulação, mostrado na Figura 3.8. A fala durante a coarticulação pode ser dividida em duas partes: banda transitória e região estável. O espectro de energia na região estável permanece basicamente invariável, e o espectro de energia na banda transitória transita suave e continuamente até o início da próxima sílaba (KANG et. Al. 2009).

Se todas as transições possíveis fossem armazenadas em um banco de dados, estas poderiam ser recuperadas de tal forma a reduzir o problema, entretanto, isto exigiria uma grande capacidade de armazenamento para o banco de dados. Uma solução alternativa consiste na modificação do espectro de energia nas transições (KANG et. Al. 2009).

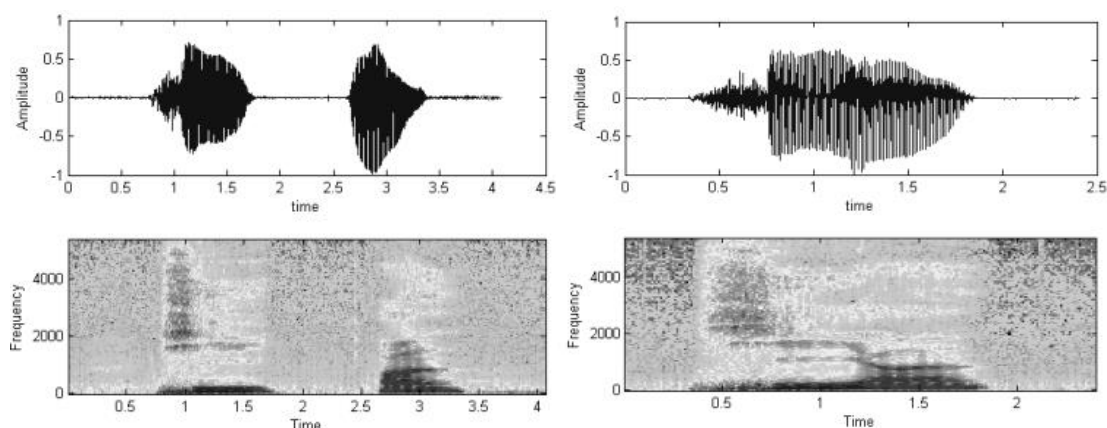



Figura 3.8: fenômeno de coarticulação para sílabas separadas (esq.) e juntas (dir.). Fonte: (KANG et. Al. 2009).

(KANG et. Al. 2009) propõe o seguinte algoritmo, mostrado na Figura 3.9, para resolver problemas de coarticulação: Aplica-se uma transformada de Fourier sobre o sinal a fim de calcular a energia desse espectro. Assim, o espectro de energia de um fonema transita suavemente para o próximo fonema por meio da modificação dos coeficientes de energia. Então, o resultado modificado sofre uma transformada inversa de Fourier, passando novamente para o domínio de tempo e então os sinais das formas de onda são concatenados por meio do algoritmo PSOLA. Ou seja, a coarticulação é sintetizada pela modificação do espectro de energia na banda transitória da fala. Tal modificação é concatenada com a região estável, seguindo então com o processo normal de concatenação de forma de onda.

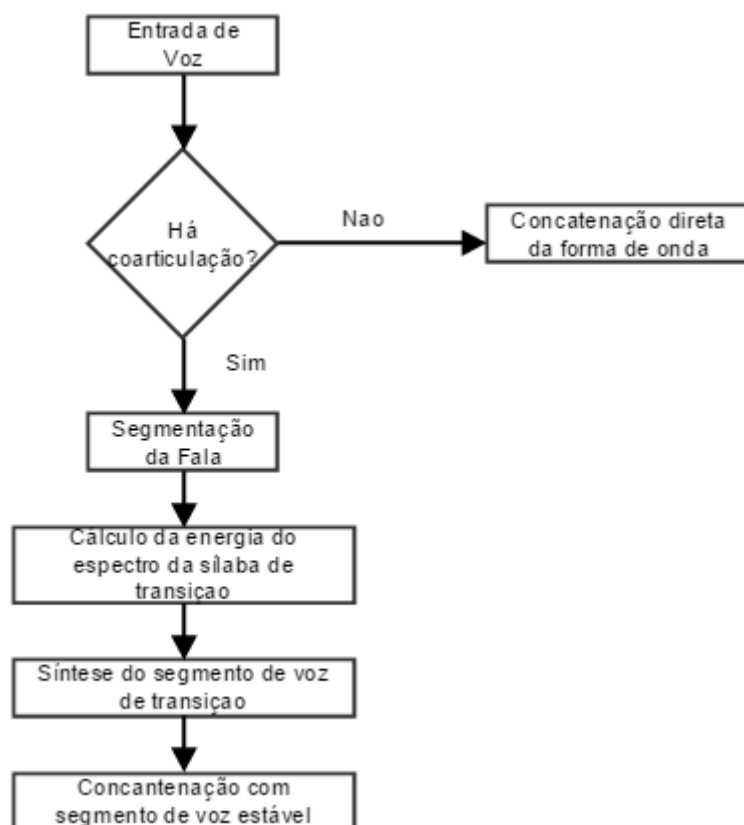



Figura 3.9: solução proposta por (KANG et. Al. 2009) para resolver problemas de coarticulação. Fonte: (KANG et. Al. 2009 - Traduzido).

No modelo coarticulatório mais básico, cada fonema tem um único "alvo" articulatório ideal para cada articulador independente dos fonemas vizinhos. Do ponto de vista coarticulatório, a transição entre dois fonemas é descrita como o movimento entre dois "alvos" ideais de dois fonemas. A transição compartilha ambas as características articulatória e acústica de ambos os alvos dos dois fonemas e gradualmente muda, estando inicialmente predominantemente semelhante ao primeiro até predominantemente semelhante ao segundo fonema-alvo posteriormente. Embora a coarticulação cause a transição na fala, trabalhos mostraram que existe um núcleo estacionário em vogais, fricativas e semi-vogais. Em tais fonemas, os núcleos são estacionários e as transições formantes entre tais fonemas, que realmente ocorrente entre os alvos nos contornos dos intervalos estacionários, são suaves (PHUNG et. al.).

Cada fonema pode ser dividido em um intervalo de núcleo e dois intervalos de transição em ambos os lados. O trabalho proposto em (PHUNG et. al.) tenta determinar as posições e durações do núcleo de dos intervalos de transição dentro de uma sílaba.

A existência de intervalos estacionários e quasi-estacionários dentro de vogais, semi-vogais e consoantes já foi demonstrada em trabalhos. A estabilidade dos intervalos

estacionários e quasi-estacionários sob efeito de coarticulação resulta que estas partes são insensíveis a contexto, de tal forma que tais partes podem ser preservadas para serem concatenadas em diferentes situações. (PHUNG et. al.) considera que as mesmas afirmações são válidas também para intervalos pseudo-estacionários

Ambos os intervalos estacionário e quasi-estacionário são considerados não sensíveis a contexto, ao contrário das outas partes. Entretanto, ainda não há métodos para se estimar a posição e a duração de cada parte dentro dos fonemas e sílabas. A decomposição temporal (TD - *Temporal Decomposition*) é um método que pode decompor a fala em componentes independentes mútuos. TD é o núcleo dos métodos propostos para modelar a coarticulação e resolver problemas de contexto em sistemas CSS (PHUNG et. al.).

3.3.6 Problemas Modificação Espectral

Algumas abordagens modificam valores espectrais dinamicamente a fim de simular a coarticulação, fazendo uso de filtros digitais com uma excitação. Outras abordagens mais simples, ao invés de armazenar padrões espectrais, armazenam formas de onda de durações variadas, concatenando-as quando necessária. Tal abordagem elimina a necessidade de filtragem. Em ambos os casos, são necessários ajustes nos contornos dos sinais (SHAUGHNESSY, 2003).

Alguns trabalhos propõem uma suavização espectral por meio de modificação das frequências formantes e da largura de banda para reproduzir a estrutura formante desejada nos pontos de concatenação. Outros propõem métodos de controlar a dinâmica espectral a fim de suavizar a trajetória das frequências formante. Métodos baseados em frames em geral tentam suavizar as descontinuidades nos pontos de concatenação, mas nenhum deles propõe corrigir de forma eficiente erros gerados por seleção inadequada de unidade em decorrência de contextos, especialmente quando os dados para concatenação são limitados. Tais problemas geralmente são gerados ou por efeitos contextuais ou coarticulação (PHUNG et. al.).

Uma vez que a síntese baseada em concatenação é limitada ao tipo de voz que foi usado na construção do banco de dados, é desejável modificar as unidades de fala a fim de remover descontinuidades e criar novas formas de fala. Entretanto, modificar a estrutura espectral geralmente conduz à degradação da qualidade do resultado. Em (WOUTERS et. al. 2000) é possível encontrar estudos sobre o uso de filtragem inversa e modelagem senoidal a fim de modificar a estrutura espectral e mantendo a qualidade da voz sintetizada. O resultado apresentou voz modificada de alta qualidade.

O modelo senoidal é uma representação atrativa da fala, porém o número de parâmetros a serem ajustados é alto e o modelo ainda não apresenta controles sobre a frequência dos formantes e largura de banda. No referido trabalho, o sinal é decomposto como soma de senos. As amplitudes complexas do modelo senoidal de um espectro discreto são aproximadas por meio de um modelo. Então é usado um modelo que se adequa à magnitude a fase e modifica a frequência dos polos e a largura do modelo. A decomposição do sinal de voz é baseado na modelagem da fala como um sinal periódico perfeito, com período do *pitch* T_0. Tal sinal corresponde a uma transformada de Fourier com valores não nulos em pontos múltiplos da frequência fundamental $f_0 = 1/T_0$. Na notação complexa, $s[n]$ é aproximado pela expressão mostrada na Equação abaixo:

$$\hat{s}[n] = \sum_{k=-L}^{L} a_k \exp(j2\pi k f_o n), \tag{10}$$

em que L é o número de harmônicos e o número complexo a_k representa a amplitude e o deslocamento de fase do k-ésimo harmônico. Note que $\hat{s}[n]$ é real se a_k e a_{-k} são complexos conjugados. A síntese senoidal pode ser realizada de diversas formas. As amostras sintetizadas $\hat{s}[n]$ podem ser calculadas usando a expressão acima enquanto se interpola os pontos entre a_k.

Uma das desvantagens do modelo senoidal consiste no fato dos parâmetros não serem diretamente relacionados às frequências formantes e largura de banda, tornando-o difícil de formular mudanças baseadas em informações a respeito dos formantes (WOUTERS et. al. 2000).

A forma mais simples de modificar a frequência fundamental é truncar cada período, removendo algumas amostras finais caso se deseje encurtar o período. Para o caso contrário, deve-se ou interpolar períodos adjacentes ou extrapolar as amostras finais (SHAUGHNESSY, 2003).

3.3.7 Marcação de Pitch

Em (KOBAYASHI et. al., 1998) é possível ver a aplicação da análise de wavelets para uma marcação de *pitch* adequada para a língua japonesa.

O sistema proposto é composto por um dicionário, cujo processo de preparação é mostrado na Figura 3.10. A preparação do dicionário se inicia com uma cuidadosa seleção dos dados de forma a assegurar um número suficientemente grande de sons para a extração dos fonemas.

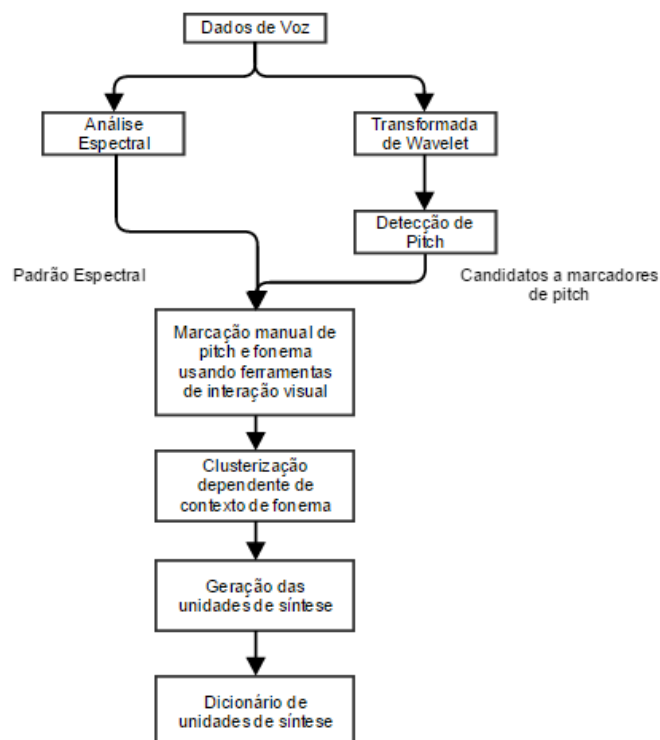



Figura 3.10: processo de preparação do dicionário para o sistema proposto em (KOBAYASHI et. al., 1998). Fonte: (KOBAYASHI et. al., 1998 - Traduzido).

No referido trabalho, propõe-se que os dados de voz sejam segmentados em fonemas por meio da observação de algumas características da fala, como espectro, dinâmica espectral e potência.

Na abordagem proposta pelo trabalho, o momento fechamento glotal é detectado por meio da busca por picos locais na transformada de wavelet da forma de onda e usa-se essa informação para a marcação do período do *pitch*. Após isso, análise espectral é usada para extração e rotulação de fonemas. O algoritmo baseado em wavelets pode ser usado tanto para vozes masculinas como femininas sem necessidade de modificar parâmetros.

A qualidade da voz sintetizada depende também do dicionário de unidades. A necessidade de um conjunto suficientemente grande para produzir uma saída de alta qualidade deve ser equilibrada com o tamanho do dicionário. Para esta tarefa, foi utilizado o algoritmo CDC (*context-dependent-culstering*), determinando o conjunto de unidades a serem instaladas no dicionário.

As três principais etapas no processo TTS desenvolvido é mostrado na Figura 3.11.

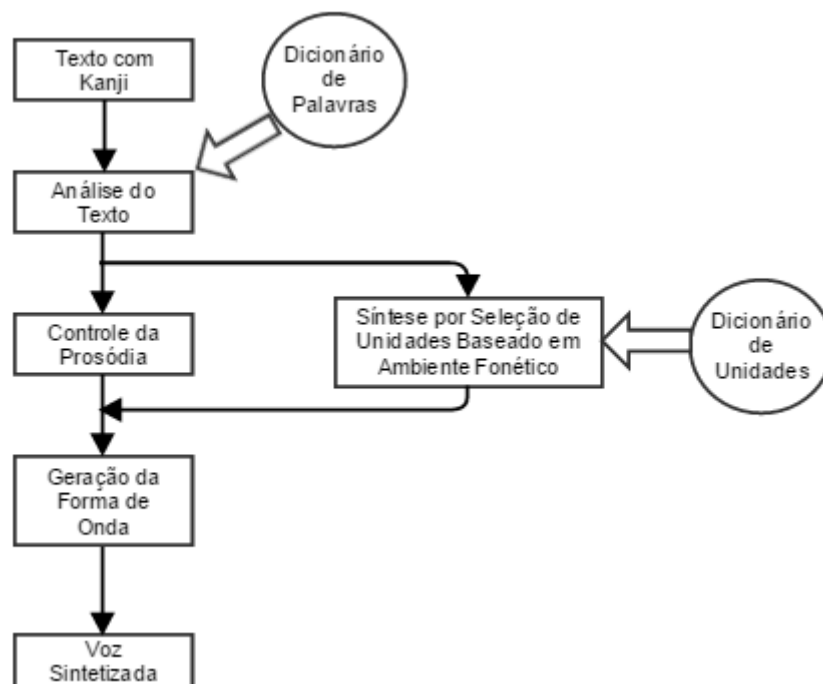



Figura 3.11: etapas principais para o processo TTS proposto em (KOBAYASHI et. al., 1998). Fonte: (KOBAYASHI et. al., 1998 - Traduzido).

A conversão TTS começa com a análise morfológica na entrada - a segmentação do texto em palavras e a análise léxica a fim de determinar a correta leitura. A segunda etapa, um *parser* é utilizado para realizar o controle da prosódia: para uma determinada frase, o sistema deve escolher um dentre quatro conexões. A etapa final usa uma versão modificada do TD-PSOLA a fim de produzir uma saída mais suave: as janelas são determinadas de forma a minimizar distorções espectrais de acordo com dois critérios: 1 - a janela de análise deve ser rigorosamente sincronizada com os instantes de excitação principal dentro de cada período de *pitch* e 2 - o sinal de voz "janelado" deve preservar as propriedades espectrais.

Marcar manualmente os períodos de *pitch*, como ocorria no algoritmo PSOLA original é impraticável para os sistemas modernos e modelos como *harmonic-plus-noise* tem sido propostos para minimizar erros de fase (SHAUGHNESSY, 2003).

3.4 Erros e dificuldades mais comuns gerados pelo processo de síntese

O maior desafio da pesquisa em síntese de voz é obter maior aproximação possível com a voz humana enquanto se minimizam os custos, sejam de memória, computacionais, treinamento, etc. (SHAUGHNESSY, 2003).

O objetivo final de um sistema de síntese de voz é não apenas produzir fala facilmente compreensível, mas indistinguível da fala humana, com o mesmo desempenho. Assim, as duas qualidades que se esperam de um sistema TTS são a inteligibilidade e a naturalidade (TABET, 2011).

Entender as limitações das soluções de acessibilidade atuais é uma das chaves para se projetar melhores *softwares* para usuários portadores de necessidades especiais.

Apesar do investimento substancial em pesquisa de tecnologias de voz nos últimos 40 anos, as tecnologias de síntese de voz ainda apresentam limitações significativas, quase sempre não atingindo a expectativa dos usuários, apresentando pronúncias inadequadas, voz pouco natural, entonação incorreta e dificuldade de reconhecer contextos, como, por exemplo, o número 110 ser sintetizado como "um um zero" ao invés de "cento e dez" ou 1kg ser sintetizado como "um k g", ao invés de "um quilo", etc. Além disso, embora existam alguns sistemas de acessibilidade e síntese de voz, a maior parte deles apresentam vozes não naturais ou não são livres.

Palavras novas, como nomes próprios de pessoas, empresas e produtos podem gerar pronúncias ambíguas, embora os sintetizadores possam pronunciar centenas ou até milhares de palavras. Pronunciar corretamente uma frase ou sentença com a melodia correta requer um entendimento do significado de uma frase que o computador não é capaz de processar, como tom de raiva, dúvida e afins, o que resulta em respostas pouco naturais, artificiais e por vezes até mesmo robóticas, pouco agradáveis de ouvir por longos períodos de tempo, o que não é desejável.

O ouvido humano é muito sensível pra pequenas mudanças na qualidade da voz. Uma pessoa pode detectar mudanças que indiquem o estado emocional, sotaques, problemas de fala, entre outros. A qualidade da síntese de voz atual ainda permanece abaixo da de uma voz real, assim, ouvintes devem fazer um esforço maior do que o normal para compreender vozes sintetizadas e devem ignorar eventuais erros. Para novos usuários, escutar uma voz sintetizada por longos períodos de tempo podem se tornar uma tarefa insatisfatória.

Assim, o desenvolvedor deve considerar duas coisas a respeito da qualidade do som: clareza e compreensão - o quanto o usuário irá entender, e naturalidade - o quanto a voz se parece com a humana. A clareza e a compreensão estão relacionadas com todas as etapas descritas no processo de síntese, uma vez que qualquer erro em uma delas poderá afetar a compreensão de modo a não se fazer entender ou ser entendido erroneamente. A naturalidade está ligada mais pelos estágios finais do processo, mais especificamente

pelo processo de métrica e geração da forma de onda (PITT, 1996; SCHUMACHER, 1995; YANKELOVICH, 1995).

É possível se ter uma voz completamente artificial e completamente compreensível bem como ter uma voz natural, mas que nem sempre seja possível entender, embora isso seja menos comum (SUN MICROSYSTEMS 1998).

Abaixo são descritos algumas situações nas quais os sintetizadores podem gerar resultados insatisfatórios.

3.4.1 Erros quanto à normalização do texto

Mudança de pronúncia de uma mesma palavra em diferentes contextos. Para este caso a solução proposta é o uso de heurísticas, estatísticas de frequência de ocorrência, examinando os vizinhos a fim de realizar a desambiguação de homógrafos.

Recentemente tem sido usados técnicas com HMM, cuja taxa de erro tem sido inferior a 5%. Converter números é um problema também frequente, pois a forma como são lidas é dependente de contextos, podendo ser lidos um a um ou como um número único. Por exemplo: 123 pode ser lido como um dois três ou cento e vinte e três. Algarismos romanos também podem ser lidos de forma diferente: enquanto "Elizabeth II" é lido como ordinal ("Elizabeth segunda"), "Capítulo II" é lido como cardinal ("Capítulo dois"). Abreviações também podem ser ambíguas. Enquanto, por exemplo, "in" pode ser abreviação para polegadas, pode ser também a preposição em inglês. Vários erros podem ocorrer também dentro do contexto de normalização do texto, como, por exemplo, os pontos na sigla "E.U.A.", que podem ser interpretados de forma errônea como fins de sentença; 1988 pode ser lido como mil novecentos e oitenta e oito ou um nove oito oito; ou ainda, construções especiais como endereços de e-mail, que são particularmente difíceis de interpretar, por exemplo: nicolas@lesc.ufc.br, pode ser lido com "nicolas arroba lesc ponto u f c ponto b r" ou "nicolas arroba l e s c ponto u f c ponto b r", uma vez não ser possível para um sintetizador conhecer todas as abreviações e acrônimos em uma língua (SUN MICROSYSTEMS 1998).

3.4.1.1 Erros na etapa de pré-processamento

As principais dificuldades encontradas nesta etapa ocorrem em situações que lidam com os seguintes tipos: números, abreviaturas e siglas.

Números são elementos frequentemente dependentes de contextos, podendo ser lidos de diversas formas, como cardinais, ordinais, datas, etc. Por exemplo: 3/4 pode significar uma fração, sendo lido como “três quartos” ou "três de abril". Além de ambiguidades de gênero: 1 pode ser lido como um ou uma. Abreviaturas são geralmente

sequências de caracteres terminados por ponto e que necessitam ser substituídos por sua forma "por extenso". Entretanto, algumas abreviaturas não são seguidas por ponto. Além disso, o número que antecede a abreviatura deverá ser colocada no plural ou no singular. Outras vezes, uma abreviação pode ter mais de uma transcrição: "cap." pode ser capitão ou capítulo, de acordo com o contexto. Siglas são sequência de letras maiúsculas delimitadas ou não por ponto. Neste caso, a dificuldade se encontra em saber se a sigla deve ser lida ou soletrada. Ademais, certos casos fogem à regra a apresentam pronúncia própria, como IEEE ("i três e") (AZUIRSON, 2009).

3.4.1.2 Erros de transcrição fonética

As principais dificuldades encontradas nesta etapa são: a determinação se as vogais "e" e "o" não acentuadas são abertas ou fechadas e a transcrição fonética da letra X. A consoante X é uma das mais problemáticas durante o mapeamento, sendo que nem sempre é possível realizar a transcrição correta por meio de regras e nesse caso, novamente deve-se lançar mão do uso de um dicionário de exceções. Ainda assim, podemos aplicar a seguinte regra, válida para boa parte dos casos: o fonema /x/ ocorre em início de palavras, depois de "n", "ai", "ei" ou "ou", o fonema /z/ ocorre em palavras iniciadas com "ex" seguido de vogal e o fonema /s/ quando seguido de consoante (AZUIRSON, 2009).

3.4.2 Erros na conversão texto-para-fonema

A síntese de voz apresenta duas abordagens básicas para a pronúncia de uma palavra, em um processo denominado conversão texto-para-fonema ou grafema-para-fonema. A abordagem mais simples é a baseada em um dicionário contendo todas as palavras e suas respectivas pronúncias armazenadas. A outra abordagem é baseada em regras de pronúncia. Cada abordagem apresenta suas vantagens e desvantagens: a abordagem baseada em dicionário é rápida e precisa, porém falha quando a palavra não se encontra no dicionário. Além disso, à medida que o dicionário aumenta, os requisitos de espaço na memória aumentam. Quanto à baseada em regras, dependendo da língua, estas podem ser muito complexas e irregulares.

3.4.3 Erros de prosódia e conteúdo emocional

Um estudo da Universidade de Portsmouth, no Reino Unido, liderado por Amy Drahota e publicado na *Speech Communication*, mostrou que ouvintes podem determinar quando um determinado locutor estaria sorrindo. A identificação das características vocais que transmitem dados emocionais pode ser usada para tornar a fala mais natural. Uma destas características é o *pitch*, que auxilia a determinar se a

frase é afirmativa, interrogativa ou exclamatória. Uma das técnicas que modificam o *pitch* envolve a transformada discreta cosseno.

3.5 Particularidades sobre a engenharia de *software* envolvendo aplicações faladas e com comandos por voz

Um fator crucial na determinação do sucesso de uma aplicação de voz é quando ou não há um benefício claro ao se usar voz. Uma interface baseada em áudio tende a ser mais agradável por simular uma conversa homem-homem, ao invés de um objeto inanimado. Entretanto, por se tratar de um meio natural de comunicação, a expectativa do usuário tende a ser extremamente alta. Isto significa que a voz é melhor usada quando a necessidade é clara, quando por, exemplo, as mãos do usuário estão ocupadas, ou quando permite que alguma tarefa seja realizada de maneira que de outra forma não seria possível, como acessar e-mails ou calendários eletrônicos pelo telefone.

Deve-se usar o reconhecimento de voz por voz quando o teclado não está disponível, as mãos do usuário estiverem ocupadas de tal forma que não seja possível usar mouse ou teclado, os comandos estão em uma estrutura de menu com muitos níveis, usuários não conseguem ou não se sentem confortáveis com digitação ou possuem algum impedimento motor. Deve-se evitar, entretanto, em ambientes muito barulhentos ou quando a tarefa for realizada mais facilmente por meio de mouse ou teclado (SUN MICROSYSTEMS, 1998).

Deve-se usar a síntese de voz quando os olhos do usuário estiverem focando sua atenção para outras tarefas mais críticas, como ao dirigir ou ao executar tarefas de manutenção ou reparo, situações que chamem atenção do usuário ou em situações em que o usuário é portador de alguma deficiência física. Deve-se evitar o seu uso quando uma grande quantidade de informações é apresentada, ao se mostrar dados que devem ser comparados ou quando a informação exposta é pessoal ou confidencial (SUN MICROSYSTEMS, 1998).

Aplicações de voz são como conversas entre o usuário e o computador. Conversas são caracterizadas por retornos verbais e não verbais para indicarem o entendimento. O maior benefício de incorporar fala em uma aplicação é que a fala é algo natural: pessoas acham falar fácil, conversar é uma habilidade que a maioria aprende desde cedo e que praticam com frequência.

Uma aplicação eficiente de voz é uma que simule alguns dos aspectos principais da conversa entre seres humanos. Interfaces bem projetadas devem se basear no

entendimento das diferentes formas da linguagem com que as pessoas se comunicam. Aplicações de voz devem adotar uma linguagem que ajude as pessoas a saberem o que elas devem fazer em seguida e tentar evitar padrões de conversação que violem a educação e o comportamento cooperativo (SUN MICROSYSTEMS, 1998).

Após definir se a fala é uma interface apropriada, deve-se considerar como a fala será integrada na aplicação. Geralmente, uma aplicação de voz é desde seu início voltada para fala. São poucas as vezes em que a fala quando acrescentada a uma aplicação pré-existente é efetiva. Traduzir uma aplicação gráfica para somente voz sem a devida adaptação também apresenta baixo índice de sucesso. As barreiras encontradas pelos portadores de deficiência visual são, em larga escala, resultado direto de produtos e serviços que não foram projetados com o intuito de serem acessíveis. A fim de reduzir essas barreiras, é necessário adicionar suporte às tecnologias assistivas.

Aplicações gráficas não são transformadas adequadamente em aplicações de fala por diversas razões. Primeiro, aplicações gráficas nem sempre refletem o vocabulário, ou até mesmo conceitos básicos, que duas ou mais pessoas usam enquanto estão falando. Por exemplo, ao se referir a um calendário, as pessoas costumam usar datas relativas, como “daqui a uma semana”, “amanhã”, “depois de amanhã”, etc.

A organização da informação é outro ponto importante a ser considerado. Apresentações que costumam funcionar bem em ambientes gráficos costumam fracassar completamente em ambientes falados. Ler exatamente o que está escrito na tela raramente é efetivo, podendo soar até mesmo estranho ao usuário. Como em um cliente de e-mail, em que, por exemplo, são mostradas informações de remetente, assunto, data e hora e tamanho. Além de tomar tempo falar todas essas informações, nem todas são necessárias, como o tamanho, e soam pouco natural. Após se ler dez mensagens, por exemplo, o usuário já esqueceu informações relevantes sobre o primeiro.

Primeiramente, é mais útil organizar os e-mails por assunto ou remetente, por exemplo. Ler esses dados são mais naturais. Por exemplo “A mensagem 2 é de Paulo Cesar Cortez , cujo assunto é entrega do artigo”. No caso de sistemas que envolvam comandos por voz também, os comandos geralmente usados em interfaces gráficas soam igualmente estranhos, como “Mover. Spam”. Embora seja um pouco mais longo dizer “Mover para a pasta spam” é mais natural, e, consequentemente, mais fácil de lembrar.

Os sintetizadores atuais ainda não soam de forma completamente natural. A escolha entre usar voz sintetizada, gravada ou simplesmente não fazer uso de recursos de voz

nem sempre é fácil. Embora uma voz pré-gravada seja muito mais fácil e agradável para o usuário, é menos efetiva quando a informação a ser apresentada é dinâmica. Usar vozes gravadas é melhor para mensagens que não mudam, enquanto voz sintetizada é melhor para textos dinâmicos.

Misturar vozes sintetizadas com gravadas, porém, não costuma trazer resultados satisfatórios. Embora, usuários relatem não gostarem de som sintetizado, elas são, de fato, eles são mais adaptáveis quando não misturados com vozes pré-gravadas. Escutar é consideravelmente mais fácil quando a voz é consistente.

Usam-se mensagens gravadas quando todo o texto a ser falado é conhecido de antemão, caso contrário, ou caso o espaço em disco seja limitado, recomenda-se o uso de sintetizadores de voz. Mensagens pré-gravadas requerem substancialmente mais espaço em disco e limitam as possibilidades de interação (SUN MICROSYSTEMS, 1998).

No contexto de inclusão digital, os requisitos de acessibilidade não devem ser um bônus disponibilizado no *software*, mas sim, colocado como prioridade, pois a acessibilidade vem sendo apoiada por leis federais e internacionais (SANTOS, 2010).

Para aplicações acessíveis, o sucesso na interação deficiente - computador consiste basicamente em ser o mais simples e amigável possível, oferecendo uma ponte através da qual as peculiaridades individuais são contempladas. Ao se desenvolver produtos voltados para deficientes visuais, o projetista deve privilegiar o uso de som, fontes com tamanho grande e, se possível, usar teclados e impressoras em Braille, monitores de tamanho maior, sensível ao toque e sistema de som completo: placa de som, microfone, caixa de som ou fone de ouvido. Ao mesmo tempo, deve-se evitar excesso de opções, uso excessivo de cores, ícones e letras pequenas e uso de mouse (SANTOS, 2010).

O desafio para desenvolvedores, que têm pouco ou nenhum conhecimento sobre questões de acessibilidade ou acerca da comunidade de pessoas com deficiências, é aprender como projetar de forma eficiente e desenvolver soluções que atendam aos requisitos necessários.

É crítico que desenvolvedores de *software* desprendam tempo projetando adequadamente aplicações voltadas para portadores de deficiência visual, uma vez que o *software* resultante pode apresentar recursos que são úteis para todos. Entretanto, para determinadas plataformas computacionais, desenvolver aplicações acessíveis pode ser um processo extremamente difícil e caro (SUN MICROSYSTEMS, 2003).

Apesar de alguns aplicativos possibilitarem que cegos utilizem programas orientados ao mouse, uma interface gráfica com vários botões ou menus numa única janela não é ótima ou eficiente para uso não gráfico. Trabalhar com interfaces gráficas ainda é mais lento e complicado para usuários com deficiência visual do que para aqueles com visão. O verdadeiro desastre ocorre quando o programa é minimizado ou sua janela perde foco por causa de outro aplicativo. Com isso, a janela se torna inacessível pelo leitor de tela até receber novamente o foco, e para o usuário, fica ainda mais “invisível”. A menos que saiba como restaurar janelas minimizadas, não fica claro para o usuário sem visão se o programa simplesmente perdeu o foco e desapareceu ou se o próprio leitor de tela travou por erro de *software*. Portanto, a interface de escolha para deficientes visuais iniciantes na computação ainda é o console de texto, que nunca perde o foco e sempre fornece um modo “tela cheia” para cada programa.

A linha de comando é a interface mais eficaz para trabalhar com computadores, pois oferece uma forma direta de introduzir comandos que fazem o computador realizar exatamente o que se deseja. Uma interface de texto direta se concentra no conteúdo, não no *layout* ou intuição visual.

3.5.1 Desafios envolvendo desenvolvimento de softwares com interface por voz

Ao desenvolver aplicações com interface por voz, que inclui tanto o reconhecimento como a síntese, depara-se com diversas peculiaridades inerentes exclusivamente a essa interface, que por vezes, tornam-se desafios e dificuldades a serem contornados pelos engenheiros de *software* e programadores.

A primeira peculiaridade é o fato de a voz ser transitória. Uma vez você ouça algo, a informação deixa de estar presente, ao contrário dos gráficos, que são persistentes. Uma interface gráfica tipicamente permanece na tela até que o usuário faça alguma coisa.

A memória de curto prazo é utilizada durante a audição. Como a voz é transitória, usuários podem lembrar apenas de um número limitado de itens de uma lista e pode acabar por perder informações importantes do começo de uma longa sentença. Por exemplo, ao falar para um sistema, o usuário frequentemente esquece as palavras exatas que falou.

Em geral transitório significa que a fala não é um meio adequado para entregar grandes quantidades de informação. Neste caso, por exemplo, listas devem ser listadas elemento a elemento em resposta ao comando “próximo” ao invés de fornecer uma lista completa (SUN MICROSYSTEMS, 1998).

Mas a natureza transitória da fala também fornece benefícios. A fala é ideal para chamar atenção ou prover um mecanismo de retorno alternativo. É possível receber notificações sem que o usuário mude de contexto de janela. Por exemplo, enquanto se trabalha na suíte de escritório, o usuário pode receber a notificação da chegada de um e-mail e pode responder, sem mudar para o cliente de e-mail, se deseja responder ou não a mensagem, ou ainda mover para a pasta spam.

Outra característica reside no fato da fala é assimétrica, ou seja, pessoas podem falar mais rapidamente e facilmente, mas nem sempre compreendem com a mesma facilidade e velocidade. Essa assimetria também significa que pessoas podem falar mais rapidamente do que digitar, mas escutar mais lentamente do que ler. Uma interface baseada em fala deve fazer o equilíbrio entre um grande número de informações para o usuário com a capacidade do usuário de absorver informações verbais.

3.5.2 Desafios envolvendo sistemas speech-only

Um sistema do tipo *speech-only* é aquele cuja entrada e saída por voz são as únicas opções de interação disponíveis para o usuário. A maioria desses sistemas são implantados na telefonia atualmente.

Em uma conversa, o tempo de reprodução é crítico. Infelizmente o atraso em decorrência do processamento em aplicações de voz frequentemente causam pausas em momentos que não são naturais. Por exemplo, o usuário responde a uma saída e por não ouvir uma resposta imediata o mesmo acredita que não se fez ouvir e repete novamente o que falou. Isso pode tanto fazer com que o usuário perca a resposta ao falar ao mesmo tempo que o dispositivo como pode causar uma falha de reconhecimento ou resposta errada.

Dessa forma, é conveniente deixar claro as seguintes informações durante a interação homem-máquina: o reconhecedor está aguardando uma resposta ou está processando a entrada de áudio? O reconhecedor ouviu o usuário? Caso afirmativo, interpretou corretamente o que o usuário disse?

É importante, em alguns momentos, realizar confirmação de ordens expressas pelo usuário seja de forma implícita, repetindo o comando entendido, ou explícita, perguntando se o usuário deseja mesmo realizar a ação que o sistema entendeu, como em caso de exclusão de dados, por exemplo.

Ao se exibir mensagens referentes a um conjunto de dados de uma mesma natureza, pode-se remover informações redundantes e/ou desnecessárias: “A temperatura em

Fortaleza é de trinta graus Celsius, no Rio de Janeiro, trinta e cinco", não sendo necessário repetir as palavras temperatura nem graus Celsius.

Na necessidade de repetir informações, pode-se fazer de forma cada vez mais curta:

"Após o bipe, grave sua mensagem e aperte parar."

"Grave sua mensagem após o bipe."

"Grave sua mensagem."

Em caso de detecção de erro, seja por parte do usuário, seja por parte do próprio *software*, é importante prover um ou mais mecanismos para correção de erros, o que nem sempre pode ser uma tarefa fácil, uma vez que o usuário tenderá a repetir a mesma frase, podendo ocasionar o mesmo erro novamente.

Neste caso, a melhor forma de lidar com isso é evitar repetir a mesma mensagem de erro. Repetições de mensagens de erro, além de não ajudarem o usuário, tendem a parecer hostis ao usuário, devendo-se então recorrer à assistência progressiva: primeiro com um "O que?", seguido de um "Desculpe, poderia repetir?" e por fim, orientar o usuário: "Tente falar pausadamente, mas sem muita ênfase".

Outra técnica é explicitar as possibilidades, do tipo sim/não, ou fornecer uma entrada de dados alternativa (SUN MICROSYSTEMS 1998).

3.5.3 Desafios envolvendo sistemas multi-modal

Sistemas multi-modais incluem outros tipos de entrada e saída além do som. No caso da latência, indicadores na interface gráfica podem indicar o estado do reconhecedor, como processando ou aguardando entrada, ao contrário do que ocorre do tipo *speech only*, além de mostrar o resultado do reconhecimento, possibilitando que o usuário veja a resposta.

Pode-se mostrar também, ao longo da etapa de processamento, os resultados preliminares da análise do que foi dito pelo usuário, que vão mudando à medida que o usuário continua a falar, ou estes podem ser ocultados ou mostrados em uma janela a parte a fim de não confundir o usuário. O que não se deve é não mostrar resultado algum, para que o usuário pense que o sistema não recebeu a entrada, facilitando a identificação de erros.

É importante ressaltar que, se a privacidade é um ponto importante, deve-se atentar para a saída não estar em volume alto.

4. TECNOLOGIAS DE SÍNTESE DE VOZ E ACESSIBILIDADE EXISTENTES NO MERCADO E O MBROLA

O presente Capítulo tem por objetivo fornecer uma visão geral sobre as principais soluções de acessibilidade e síntese de voz disponíveis no mercado atualmente, citando suas características, vantagens e desvantagens. Além disso, é apresentado o método de síntese de voz baseado em concatenação de unidades sonoras e o MBROLA, sistema baseado em concatenação e parte integrante do sistema desenvolvido. Para maiores informações referentes a outras técnicas de síntese de voz, consultar os apêndices referentes à modelagem matemática do trato vocal e algoritmos de síntese de voz.

4.1 Sistemas de acessibilidade e síntese de voz existentes no mercado

Desde o início da computação orientada ao mouse, o *desktop* gráfico foi projetado para usuários que trabalham dentro de um contexto visual. Entretanto, atualmente têm surgido soluções que visam mudar essa situação na tentativa de garantir aos portadores de deficiência visual o acesso aos recursos de informática.

As características gerais de alguns dos sistemas de acessibilidade e síntese de voz mais usados ao redor do mundo, inclusive no Brasil, são descritas a seguir, destacando suas vantagens e desvantagens. É interessante observar que as soluções apresentadas, em geral, não apresentam voz natural, não são nativamente multiplataformas, apresentam suporte para um número restrito de idiomas e / ou não são livres ou custam valores elevados para camadas populares. Além disso, a maioria apresenta apenas sintetizador de voz, não provendo um pacote contendo as aplicações mais usadas no dia-a-dia de um usuário comum.

Vale ressaltar ainda que esta lista apresenta apenas algumas das soluções existentes, e não todas, deixando de lado *softwares* como o JAWS e Virtual Vision, que custam, respectivamente, US$ 1.200,00 e US$ 2.500,00 e são disponíveis apenas para plataforma Microsoft Windows, tornando-se proibitivos para usuários que não disponham de tais recursos financeiros ou não usem esta plataforma.

Vale ressaltar que, excetuando-se o ADRIANE, DOSVOX/LINVOX, LianeTTS e NVDA, os demais sistemas são apenas sintetizadores de voz, não englobando soluções de acessibilidade. Ademais, são poucos os sistemas que realizam uma análise semântica-pragmática de textos e quando o fazem, é comum os algoritmos do *parser* produzirem estruturas sintáticas incorretas (AZUIRSON, 2009).

4.1.1 Acapella

Acapella TTS é um sintetizador de voz projetado para desenvolvedores integrarem a capacidade de síntese de voz para suas aplicações nos dispositivos baseados em GNU/Linux embarcado. Apresenta uma das sínteses com maior qualidade já existentes, com SDK disponível para teste. Apresenta suporte para até 33 línguas, 100 vozes e plataformas ARM, MIPS e Intel x86, entretanto, não se trata de uma solução livre e, como dito, voltado apenas para sistemas GNU/Linux (ACAPELLA 2014).

4.1.2 ADRIANE

O projeto ADRIANE (*Audio Desktop Reference Implementation And Networking Environment* – Ambiente de Rede e Referência para Implementação de *Desktop* Auditivo) visa prover uma interface de usuário passo-a-passo e linear, fácil de usar e organizada em menus que priorizam os aplicativos e tarefas mais usadas pelo usuário. A primeira linha do ADRIANE diz "Enter para ajuda, seta para baixo o próximo menu". O sistema contém leitores de tela, sintetizadores de voz, drivers Braille, navegação pelo teclado e programas que podem ser inteiramente utilizado por meio de interações não gráficas. Além disso, com o GSM, o usuário do ADRIANE consegue baixar mensagens SMS para o computador e respondê-las com uso de um editor e um teclado normal, em vez de pequenas telas do telefone.

A equipe do ADRIANE procurou desenvolver *softwares* que se adaptassem às capacidades e limitações dos usuários, ao invés de adaptar uma interface pré-existente cujo desenvolvimento inicial não previa oferecer suporte aos deficientes visuais. A pedido especial de usuários e programadores cegos mais experientes, depois foi acrescentado um item *Shell* ao primeiro menu.

O sistema ADRIANE está disponível no Live CD ou DVD do Knoppix desde a versão 5.3 por meio da opção de inicialização Adriane. Também é possível remasterizar o CD ou DVD para usar o ADRIANE como opção padrão.

Como desvantagem, pode-se afirmar o fato do projeto ADRIANE ser exclusivo para ambiente GNU/Linux, que restringe o campo de uso para apenas os usuários deste sistema operacional (KNOPPER, 2009).

4.1.3 Aiuruetê

Iniciado em 1991 pelo Laboratório de Fonética e Piscolinguística (LAFEPE) em conjunto com o Instituto de Estudos da Linguagem (IEL) da Universidade Estadual de Campinas (UNICAMP). Trata-se de um projeto acadêmico baseado em síntese concatenativa de polifones capaz de diferenciar maior ou menor abertura vocálica por

meio da identificação da classe gramatical. O sistema foi desenvolvido em C++ e Delphi e é voltado apenas para plataforma Microsoft Windows (AZUIRSON, 2009).

4.1.4 DOSVOX e LINUXVOX

De acordo com o manual de usuário do sistema, o DOSVOX é um sistema para microcomputadores da linha PC que se comunica com o usuário mediante síntese de voz, viabilizando o uso de computadores por deficientes visuais. O programa é composto de: "sistema operacional" que contém os elementos de interface com o usuário, sistema de síntese de fala para língua portuguesa, editor, leitor e impressor/formatador de textos, impressor/formatador para Braille, programas de uso geral adaptado a cegos, como agenda, calculadora, jogos, ampliador de telas para pessoas com visão reduzida, programas educacionais para crianças, clientes para acesso à internet, como cliente de correio eletrônico, Telnet, FTP, páginas Web, aplicativos multimídia, leitor de telas para Windows, etc.

O sistema foi desenvolvido pelo Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro, sob a supervisão do prof. Antônio Borges, da Divisão de Assistência ao Usuário, em conjunto com Marcelo Pimentel. Da equipe de desenvolvimento participam também programadores deficientes visuais (AZUIRSON, 2009).

Ao contrário do que consta no manual, o DOSVOX não é um sistema operacional, uma vez que necessita de uma plataforma operacional para ser executada e não é responsável por tarefas de gerenciamento de *hardware* - incluindo memória e E/S, processos ou sistemas de arquivos. O sistema em sua maior parte é baseado em vozes pré-gravadas - o que limita as possibilidades de interação com o sistema, portanto não é síntese em si: o DOSVOX não realiza processamento linguístico nem processamento prosódico (AZUIRSON, 2009).

Por ter sido desenvolvido em Pascal, não faz proveito da portabilidade oferecida pelo Java. O sistema foi desenvolvido nativamente para Windows, e embora possa ser usado por meio do Wine no GNU/Linux, tal solução pode tornar o sistema instável.

O projeto gratuito LINVOX é a implementação do DOSVOX em Linux, ao executar o mesmo no referido ambiente usando o Wine e contém um pacote de aplicativos nativos *open source* que possibilitam acessibilidade ao Linux baseado no DOSVOX. O sistema conta com um sintetizador de voz em português e um leitor de tela *open source* - devido à grande tendência na época de sua criação na utilização do

Linux, que possui acesso completo ao ambiente gráfico do Linux, funciona em modo texto e gráfico e compatível com várias distribuições.

O projeto tem como objetivo facilitar a produção cultural de portadores deficientes visuais, permitir a alfabetização em todos os níveis, fundamental, médio e superior, e fornecer suporte às profissões já existentes.

4.1.5 eSpeak

O eSpeak, mostrado na Figura 4.1, é um *software* sintetizador de voz para inglês e outras línguas, incluindo português brasileiro, para GNU/Linux e Microsoft Windows. O sistema provê um programa em linha de comando que gera falas a partir de textos ou entradas-padrão e bibliotecas compartilhadas por programas, como por exemplo as DLLs do Microsoft Windows.

Como é possível perceber, não se trata de um sistema de acessibilidade propriamente dito, sendo apenas um sintetizador de voz, não possuindo uma interface que permita a interação direta com um usuário deficiente visual.

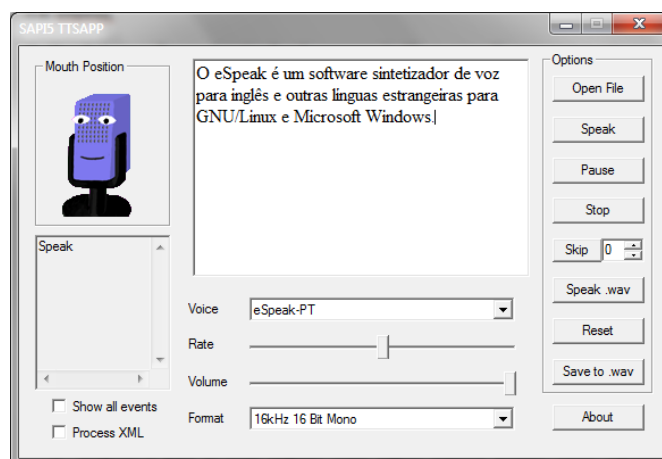



Figura 4.1: interface gráfica do eSpeak.

Dentre suas principais características, pode-se citar: disponibilidade para diversas plataformas como Android, Mac OSX e Solaris; apresentando alta compatibilidade com o sistema ADRIANE; disponibilidade de diferentes vozes, cujas características podem ser alteradas; possibilidade de produzir saída no formado WAV; suporte para HTML; tamanho compacto; possibilidade de ser utilizado como *front-end* para o MBROLA, porém, não é acessível, não fornece pacotes de softwares acessíveis e não possui possibilidade de atuar como *front-end* para outras *engines*; escrito em linguagem C e

disponível para mais de 30 idiomas, incluindo inglês, francês, alemão, russo, espanhol e inclusive português brasileiro (ESPEAK, 2014).

4.1.6 Festival

O Festival é um sistema TTS desenvolvido inicialmente pela Universidade de Edimburgo, sendo um *front-end* para o MBROLA e outras *engines*, não possuindo um cliente TTS *stand-alone*. Possui uma versão em português não livre baseado em síntese de formantes (COSTA e MONTE, 2012).

Festival oferece um *framework* geral para o desenvolvimento de sistemas de síntese de voz por meio de APIs, interpretadores de comando, bibliotecas em C++ e Java, e interface para o Emacs. Está disponível em inglês, britânico e americano, e espanhol. O sistema é escrito em C++ e usa a biblioteca *Edinburgh Speech Tools*. Trata-se de um *software* livre, distribuído sob licença X-11 permitindo uso irrestrito comercial e não-comercial. A versão estável atual é a 2.1 e apresenta as seguintes características (FESTIAL, 2014): integração de API de síntese baseado em modelos de Markov; suporte para GCC 4.3, 4.4 e 4.5; suporte a áudio nativo do Apple OS X; retrocompatibilidade com Festival 1.4.3; suporte à base de dados do MBROLA.

O Festival é tido como um sistema de síntese de voz para pelo menos três níveis de usuários. No primeiro nível, é destinado para aqueles usuários que simplesmente querem uma alta qualidade de voz de textos arbitrários com o mínimo de esforço. No segundo, é dirigido para aqueles que estão desenvolvendo sistemas de idioma e desejam incluir saída sintetizada. Neste caso, é desejado e necessário uma certa quantidade de padronização, assim como vozes diferentes, etc. O terceiro nível consiste em desenvolver e testar novos métodos de síntese.

A filosofia adotada por sistemas como o Festival permite a adição e teste de novos módulos de voz sem a necessidade de gastar esforços significativos para construir um sistema inteiro ou adaptar um já existente.

Este é um sistema TTS inteiramente apropriado para ser utilizado em outros projetos que necessitem de saída de voz, além disso, pode-se identificar três partes básicas do processo TTS: *Text Analysis*, *Linguistic Analysis* e *Waveform Generation*.

O processo *Text Analysis* (Análise do Texto) tem como propósito colocar e organizar as orações em uma lista de gerenciamento de palavras, identificar números, abreviações e acrônimos, transformando-as em texto por extenso, por exemplo: “Sr.” é transformado em “Senhor”, quando necessário, utilizando uma gramática regular como base para solucionar alguns problemas. Também é responsável por determinar a classe

de casa palavra, individualmente, analisando a ortografia das mesmas e organizando uma lista de categorias e fazer a flexão e a derivação das palavras, quando necessário, decompondo-as em unidades gramaticais elementares através da análise de suas raízes léxicas e seus afixos - prefixos e sufixos. Além disso, analisa as palavras observando o contexto em que estão inseridas, ou seja, analisando a palavra em questão associada aos seus vizinhos, possibilitando assim uma melhor identificação e diminuição da lista de categorias.

Linguistic Analysis é o processo responsável pelo gerenciamento e produção da prosódia utilizada na geração dos sons. Conforme dito anteriormente, a prosódia se refere a certas propriedades de sinais da fala que estão relacionadas à mudança de entonação da voz, sonoridade e duração do som das sílabas. A prosódia influi diretamente na comunicação por voz e tem uma função bastante específica e fundamental nesse tipo de comunicação.

Por fim, o processo *Waveform Generation* é responsável pelo controle dinâmico de articulações e controle da frequência vibratória das dobras vocais, que possibilitam a produção de sinais de voz exigidos.

O Festival está em constante desenvolvimento e pretende incluir diversos outros módulos. Aperfeiçoamentos já estão sendo considerados em vários estágios de implementação, como técnicas podem-se citar síntese baseada em seleção, especificação léxica independente do dialeto, dentre outras (FESTIVAL, 2014).

Entretanto, criar um banco de dados e um conjunto de regras de fala para o Festival não é fácil, pois usa uma sintaxe semelhante à linguagem de programação Lisp e requer um banco de dados de dífonos com aproximadamente 3 mil trechos de áudio, cortados e estendidos por pontos de entonação. Há somente algumas poucas vozes gratuitas gravadas para o Festival no momento, dificultando seu empego em larga escala ou a sua popularização em massa.

4.1.7 FreeTTS

FreeTTS é um sistema de síntese de voz escrito inteiramente em Java. Free TTS inclui uma *engine* de síntese de voz com suporte para vozes, masculina em inglês americano de 8 e 16 KHz e para voz MBROLA masculina e feminina a 8 e 16 Khz e suporte para importar vozes do FestVox. Além disso, possui compatibilidade parcial com JSAPI e ampla documentação incluindo diversas aplicações demonstrativas. Apesar da facilidade de uso, não apresenta suporte para português (FREE TTS, 2014).

4.1.8 Furbspeech

O TTS Furb-Speech foi um *front-end* para o MBROLA desenvolvido em Java pela Faculdade de Blumenau. Aparentemente o projeto foi descontinuado, pois a última atualização do projeto foi realizado em 2009, não sendo integrado também a nenhum sistema de acessibilidade (COSTA e MONTE,2012).

4.1.9 IBM Via Voice

O IBM Via Voice é uma plataforma proprietária - o que impede que o usuário adapte o programa conforme suas necessidades, não só de síntese, mas também de reconhecimento de voz. Voltado também para sistemas embarcados, apresenta versões para Microsoft Windows e Mac OS X. A última versão estável foi a 9.0 Em 2003, a IBM vendeu o ViaVoice para a ScanSoft, sendo agora chamado Nuance (IBM VIA VOICE, 2015).

Sua tela principal é mostrada na Figura 4.2. Observa-se que as línguas são limitadas, não contemplando a língua portuguesa, entre outras.

Figura 4.2: IBM Via Voice.

Posteriormente, surgiu o Projeto Voxin, uma parceria com a IBM, para a aquisição do sistema TTS IBM ViaVoice, que é um sistema TTS não livre que pode ser usado em diversas aplicações, como leitores de tela, de boa qualidade, podendo ser também integrado a ferramentas e sistemas operacionais livres (COSTA e MONTE, 2012).

4.1.10 Liane TTS

O LianeTTS é um compilador que analisa texto e o traduz em texto compilado no formato de dífonos para processamento e síntese de voz pelo MBROLA. Este realiza a tarefa de concatenar dífonos. Além disso, consiste em um *front-end* para o MBROLA e

scripts para integração ao leitor de tela ORCA por meio do driver *speech-dispatcher* e incluiu ao MBROLA uma voz feminina chamada br4 (COSTA e MONTE, 2012).

O LianeTTS passou a ser utilizado em larga escala em info-centros através de projetos governamentais de inclusão digital (COSTA e MONTE, 2012).

O LianeTTS é uma aplicação de *software* livre voltado para o sistema operacional GNU/Linux, permitindo que deficientes visuais utilizem computadores. O sistema é escrito em linguagem C e produz síntese de voz em Português do Brasil com sotaque carioca, tendo sido produzido pelo Serviço Federal de Processamento de Dados (SERPRO) e do Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro (NCE/UFRJ). Apesar dos esforços, o LianeTTS não tem recebido boas críticas por parte de seus usuários (LIANETTS, 2014).

4.1.11 Nambiquara

Trata-se de um sistema TTS livre, baseado em síntese concatenativa, servindo de *front-end* para o MBROLA, sendo programado em PHP sobre um servidor apache, auxiliado por formulários HTML e scripts em JavaScript com banco de dados de siglas desenvolvido em MySQL. Como em quase todos os sistemas concatenativos, apresenta grande dificuldade para modelar características emocionais e dependentes de contexto, realizando uma fala sempre neutra (AZUIRSON, 2009).

As etapas de processamento realizados pelo Nambiquara são mostrados na Figura 4.3.

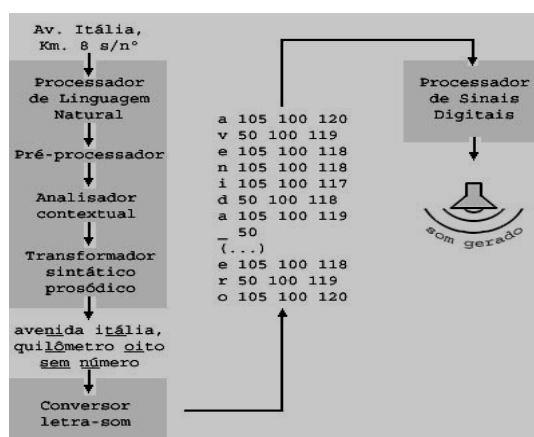



Figura 4.3: diagrama de Blocos do Nambiquara. Fonte: (AZUIRSON, 2009).

4.1.12 NVDA

NVDA (*Non Visual Desktop Access*) é um leitor de telas disponível para 48 línguas, livre e de código-fonte aberto, sob Licença GNU, voltado para a plataforma

Microsoft Windows. Foi criado por Michael Curran em 1996, sendo desenvolvido em Python e uma parte em C++ e baseado nas APIs *Microsoft Active Accessibility*, *IAccessible2* e *Java Access Bridge*.

O NVDA utiliza o eSpeak como sintetizador integrado e provê suporte a aplicações como WordPad, Notepad, Internet Explorer, Google Chrome, Outlook Express, Mozilla Thunderbird, Microsft Word, Microsoft Excel e Microsoft PowerPoint. Por meio do *Java Access Bridge*, provê suporte também ao LibreOffice e OpenOffice.

4.2 O MBROLA

O objetivo do projeto MBROLA, mostrado na Figura 4.4, iniciado pelo laboratório CTS da *Faculté Polytechnique de Mons*, na Bélgica, é obter um conjunto de sintetizadores de voz para a maior quantidade de línguas possível e disponibilizá-las para aplicações livres não comerciais e não militares, além de impulsionar pesquisas sobre síntese de voz, particularmente, geração de prosódia, um dos maiores desafios atuais a respeito da síntese de voz (MBROLA, 2014).

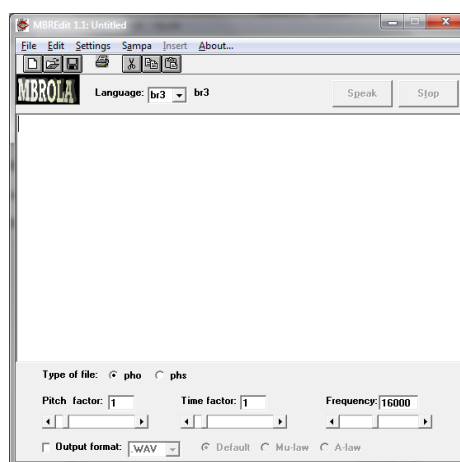



Figura 4.4: interface Gráfica do MBROLA.

O ponto central do MBROLA é o um sintetizador baseado na concatenação de dífonos que usa como entrada uma lista de fonemas juntamente com informações sobre prosódia, como duração dos fonemas e tom, e produz amostras de 16 bits. Assim, por não aceitar como entrada texto puro, o MBROLA não é considerado um sistema TTS (MBROLA 2014; DUTOIT 1993; DUTOIT 1997).

O projeto MBROLA está disponível para diversas plataformas como Microsoft Windows, GNU/Linux, MacOSX, NetBSD, FreeBSD, Solaris, BeOS, QNX, Symbian, etc.

Oficialmente, o projeto MBROLA disponibiliza 3 vozes diferentes para o português brasileiro: br1, b2 e b3, sendo todas as três masculinas. Um grupo liderado por pesquisadores do SERPRO e da UFRJ disponibilizaram recentemente uma voz feminina denominada br4 (COSTA e MONTE, 2012).

4.2.1 O Algoritmo

O MBROLA (*Multi Band Resynthesis OverLap Add*) é um algoritmo para síntese de voz no domínio do tempo baseado em dífonos que usa uma variante do método PSOLA, uma patente da France Telecom e permite uma grande qualidade no som gerado. Assim como ocorre no PSOLA, há um baixo custo computacional. Entretanto, ao contrário do PSOLA, o MBROLA não exige marcação preliminar de períodos de *pitch*. Embora seja baseado em dífonos, a qualidade da síntese do MBROLA é considerada superior aos demais sintetizadores baseados nesta técnica uma vez que há um pré-processamento dos dífonos impondo fases de modificação de tom e harmônicos a fim de melhorar a concatenação. O MBROLA dispõe de um grande banco de dados contendo conjuntos de dífonos para diversas línguas e vozes, auxiliado por empresas, laboratórios e voluntários ao redor do mundo, embora ainda tenha algumas línguas importantes em falta como o chinês. É um sistema muito rápido e que usa pouca memória, sendo adequado para execução em máquinas modestas, ou em ambientes com grande quantidade de sínteses de voz por segundo. O arquivo de extensão .pho usado como entrada pelo MBROLA contém uma lista de dífonos a serem concatenados, contendo informações com nome dos fonemas, duração em milissegundos e curva de prosódia contendo posição em porcentagem e *pitch* (MBROLA 2014; DUTOIT 1993; DUTOIT 1997).

O formato de um arquivo .pho para a palavra noite é mostrado na Figura 4.5.

Figura 4.5: formato de um arquivo .pho para a palavra "noite".

O MBROLA, através de uma lista de fonemas de entrada em conjunto com dados de prosódia (*pitch* e duração de fonemas em milissegundos), gera vozes de 16 bits e pode gerar arquivos .wav, au, raw e aiff (AZUIRSON, 2009).

Os pontos de *pitch* são determinados pela posição relativa em percentual da mudança da entonação e o *pitch* em Hertz (AZUIRSON, 2009).

O MBROLA Faz uso de um banco de dados de dífono especialmente adaptado aos requisitos do sintetizador após passar por um processo de análise/síntese harmônico/estocástico a partir de um banco de dados de dífono original, um banco de dados composto por amostras, tirando vantagem da flexibilidade do modelo paramétrico enquanto que mantém a simplicidade computacional dos modelos no domínio do tempo. O algoritmo apresenta baixo custo computacional, com 7 operações por amostra em média enquanto permite ao sintetizador uma suavização espectral no domínio do tempo nas vizinhanças do segmento, tornando o resultado mais fluido (MBROLA 2014).

A Figura 4.6 mostra o diagrama de blocos do MBROLA de forma detalhada. O algoritmo MBROLA recebe como entrada informações fonéticas e prosódicas que são utilizadas como entrada para o Gerador de Lista de Segmentos. Este último também faz uso de segmentos de voz pré-gravados e armazenados em um banco de dados. Tais segmentos sofrem um processo de compressão e codificação quando armazenados e descompressão, equalização para a prosódia correspondente e decodificação. Por fim, tais segmentos são concatenados, gerando uma fala sintetizada, que é a saída do sistema.

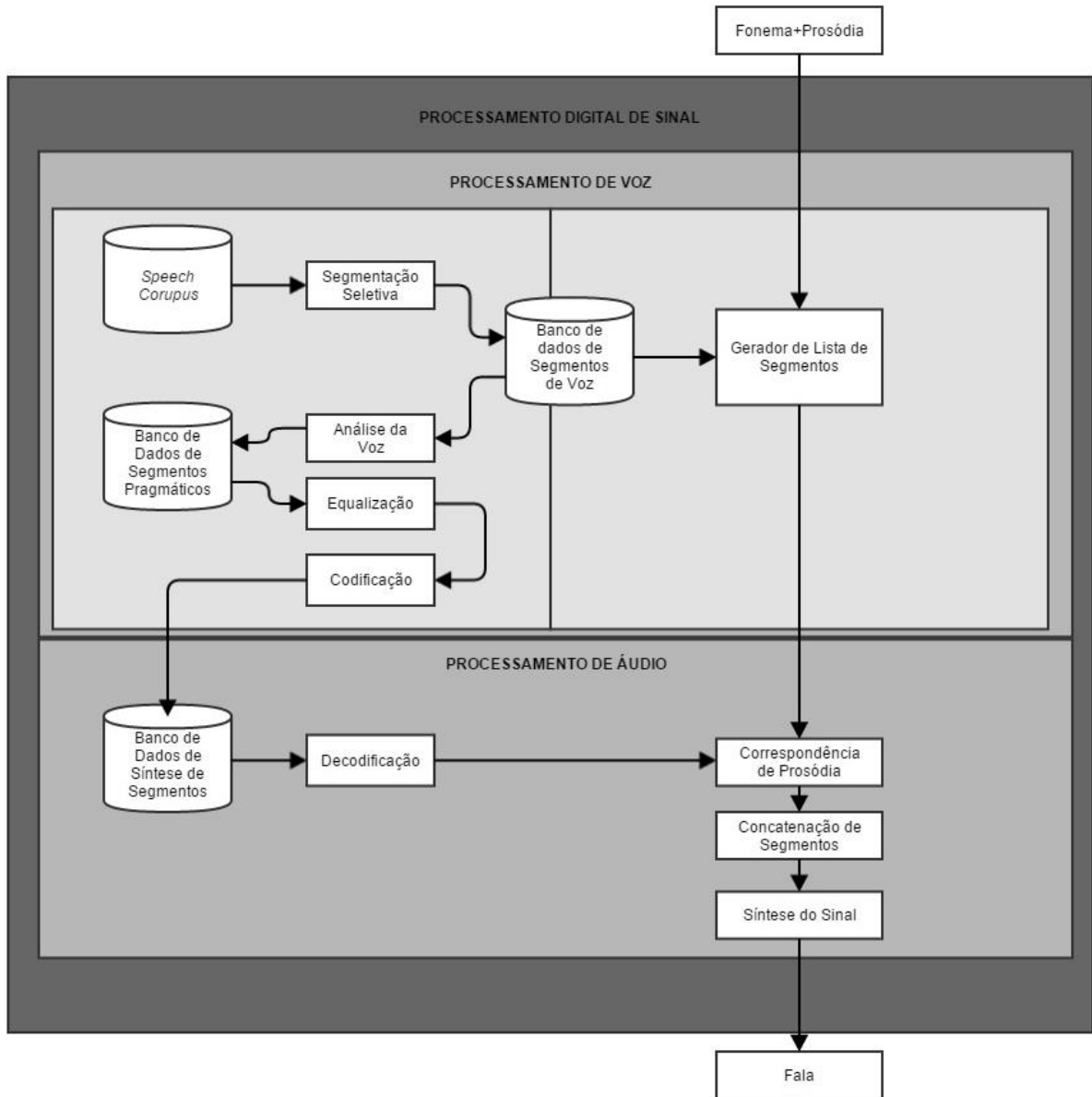



Figura 4.6: diagrama esquemático para o MBROLA.

O processo de síntese de voz pode ser modelado pelas Equações 11, 12 e 13:

$$w_j(n) = w_1\left(\frac{nT_e}{T_{j_0}}\frac{1}{F_R}\right), \tag{11}$$

$$s_j(n) = s(n)w_j\left(n\eta^j\right), \tag{12}$$

$$\hat{s}(n) = \sum_{j=-\infty}^{\infty} s_i\left(n-\eta^j\right), \tag{13}$$

em que w_1 é um valor de peso que varia dentro do intervalo [0 1] e η^j é um valor denominado *pitch marker*. F_R tem valor padrão unitário. Nesse caso, o somatório possui no máximo quatro termos, com o fator de *pitch* – a razão entre período de pitch de

síntese local e o original - $F_P = \frac{T}{T_0}$, variando no intervalo [0,5 2]. A precisão de aproximação depende do valor do período do *pitch*. Quando $F_P > 1$, a síntese tende a falhar, se $F_P < 1$, o valor de K se torna altamente dependente do fator de *pitch*. Em 1989 foi proposto um modelo que propõe que cada amostra de síntese seja multiplicado por dois fatores de normalização. Tal modelo é mostrado na Equação 14.

$$\hat{s}(n) = \sum_i \alpha_i \frac{s_i(n - \eta^i)}{w_i(n - \eta^i)}, \qquad (14)$$

em que α_i é introduzido para compensar a dependência de K em F_P e o denominador atua como fator de compensação dinâmica que contrabalanceia as variações de K com n. Entretanto, trabalhos publicados por Dutoit mostraram que não há degradação significante quando o denominador não é usado, eliminando-o e adotou também $\alpha_i = \frac{1}{F_P}$ (DUTOIT, 1997).

O MBROLA e o TD-PSOLA podem ser vistos então como intermediários entre duas situações extremas, nas quais nenhum deles oferece resultados de síntese satisfatórios: se F_R é muito grande, as linhas espectrais aparecem no espectro de $s_i(n)$, o que evita a reharmonização de $s(n)$, se F_R é muito pequeno, uma harmonização grosseira será produzida. Além disso, a aproximação não será válida. O caso intermediário fornece uma qualidade muito boa para certos valores de F_R: se $F_R \approx 1$, o espectro de $s_i(n)$ aproxima-se do envelope do espectro de $s(n)$ e a operação de reharmonização altera o *pitch* sem afetar as frequências formantes e a largura de banda (DUTOIT, 1997).

Sinal reconstruído é aproximadamente igual ao sinal da voz humana original:

$$\tilde{s}(n) = \sum_{i=-\infty}^{\infty} s(n)w(n - iT) = s(n)\widehat{w}(n) \approx Ks(n), \qquad (15)$$

em que $w(n)$ é denominada janela de ponderação. Para o caso de uma janela triangular, com tamanho igual ao dobro do período do *pitch*, temos uma redução na expressão para uma igualdade exata com $K = 1$. No caso particular do MBROLA e do TD-PSOLA, a lista de parâmetros se reduz a sequências de marcadores η^i indicando o centro de quadros OLA. Eles são posicionados de forma sincronizada com o *pitch* nas partes vozeadas de segmentos por meio de auxílio através de um algoritmo de extração de *pitch*, e igualmente espaçados nos trechos sem voz. Na prática, o comprimento da janela de ponderação w(n) é implicitamente adaptado do período do *pitch* local, assim as

amostras $s_i(n)$ diferem de zero apenas em um intervalo que depende do fator de sobreposição F_R, definido como a taxa do tamanho L da janela $w(n)$ pelo período do pitch de análise menos um ($F_R = \frac{L}{T_0} - 1$) (DUTOIT, 1997).

O comparativo entre algumas das soluções de síntese de voz disponíveis atualmente é mostrado na Tabela 4.1.

Tabela 4.1: comparação entre as diversas plataformas de acessibilidade e sintetizadores de voz existentes.

	Tipo	**Suporte à Língua Portuguesa?**	**Licença Livre?**	**Gratuito?**	**Linguagem de Programação**	**Sistema Operacional**
DOSVOX	**Vozes Pré-Gravadas**	Sim	Sim	Sim	**Pascal**	**Windows**
LINVOX	**Vozes Pré-Gravadas**	Sim	Sim	Sim	**Pascal**	**Linux**
ADRIANE	**Síntese**	**Não**	Sim	Sim	**C**	**Linux**
JAWS	**-**	**Não**	**Não**	**Não**	**-**	**Windows**
Virtual Vision	**-**	**Não**	**Não**	**Não**	**-**	**Windows**
MBROLA	Síntese	Sim	Livre com restrições	Sim	**C**	Windows, Linux, MacOS, etc.
Festival	Síntese	**Não**	Sim	Sim	**C++**	Windows, Linux e Mac OS
IBM Via Voice	Síntese	Sim	**Não**	Sim	**-**	Windows e Mac OS
eSpeak	Síntese	Sim	Sim	Sim	**C**	Windows, Linux, Solaris, Android e Mac OS
Acapella	Síntese	Sim	**Não**	**Não**	**C**	**Linux**
Liane	Síntese	Sim	Sim	Sim	**C**	**Windows e Linux**

Fonte: Próprio autor.

5. SISTEMA DESENVOLVIDO

O presente Capítulo visa apresentar o sintetizador de voz proposto, citando suas características gerais, suas vantagens, metodologia de desenvolvimento, detalhando seu funcionamento e as ferramentas acessíveis desenvolvidas.

O projeto desenvolvido é um *front-end* para o MBROLA desenvolvido em Java a fim de poder ser executado em diferentes ambientes operacionais, com GNU/Linux, Mac OS e Microsoft Windows, por exemplo. O projeto segue o modelo de *software* livre e gratuito - em oposição a algumas das interfaces atuais que são fechadas e apresentam alto custo financeiro. Este projeto é, portanto, de código aberto e de livre distribuição para que os interessados possam fazer modificações e uso de acordo com suas necessidades, facilitando e acelerando ainda mais o processo de inclusão digital de deficientes visuais. Apesar da existência de vários sistemas de acessibilidade e síntese de voz, a maioria deles apresenta uma série de desvantagens como já foi abordado.

5.1 Teste de diálogo natural

A fim de assegurar a qualidade do *software*, após determinar os requisitos do sistema, foram iniciados os primeiros testes. Aplicações acessíveis requerem testes especiais a fim de garantir se o mesmo atende às especificações. O primeiro teste é o chamado "Estudo de Diálogo Natural": dois usuários devem completar uma determinada tarefa. Um usuário deve possuir um computador e um telefone. O outro deve possuir apenas um telefone. O primeiro representa o *software* e o segundo representa o usuário final do produto. Então, deve-se observar o diálogo entre as duas pessoas, analisando as ordens dadas pela pessoa que representa o usuário e as mensagens fornecidas pela pessoa que representa o sistema. Esta técnica é utilizada para coletar vocabulário e estabelecer um padrão de gramática, fornecendo ideias para mensagens e respostas. Trata-se de um teste barato, rápido e que não requer um grande número de pessoas e muito menos uma implementação prévia do sistema. Uma versão mais sofisticada do teste envolve vários voluntários atuando como usuários. Com base nessa análise, foi desenvolvida uma interface linear e objetiva, semelhante à existente no ADRIANE, em que são apresentadas inicialmente as opções de programas disponíveis e letras de atalho no teclado correspondentes, com a opção para repeti-las sempre que se deseje.

5.2 As ferramentas utilizadas

5.2.1 Linguagem JAVA

A linguagem Java foi escolhida por ser uma linguagem de alto nível orientada a objeto, que apresenta uma ampla documentação e uma grande variedade de APIs e frameworks para as mais diversas aplicações e que é constantemente atualizada. Entretanto, o principal motivo pelo qual a linguagem foi escolhida foi pela portabilidade oferecida pela linguagem.

Devido à existência da JVM (*Java Virtual Machine*), que atua como uma camada de *software* entre o sistema operacional e a aplicação desenvolvida, não há necessidade de recompilar o projeto para cada plataforma operacional, uma vez que a JVM é responsável por fazer a ponte entre os *bytecodes* Java e o sistema operacional.

5.2.2 IDE NetBeans

O Netbeans, mostrado na Figura 5.1, é um ambiente de desenvolvimento integrado livre, gratuito, com suporte a diversas linguagens, como Java, C/C++ e PHP, e disponível para plataformas Microsoft Windows e GNU/Linux. Seu editor de código-fonte possui destaque de sintaxe, destaque de elementos selecionados, fechamento automático de delimitadores, identação automática, auto completar, marcação de *imports* não utilizados e integração com Javadoc. Além disso, possui designer de interface gráfica, *debugger*, recursos de refatoração, suporte a controle de versão e JUnit.

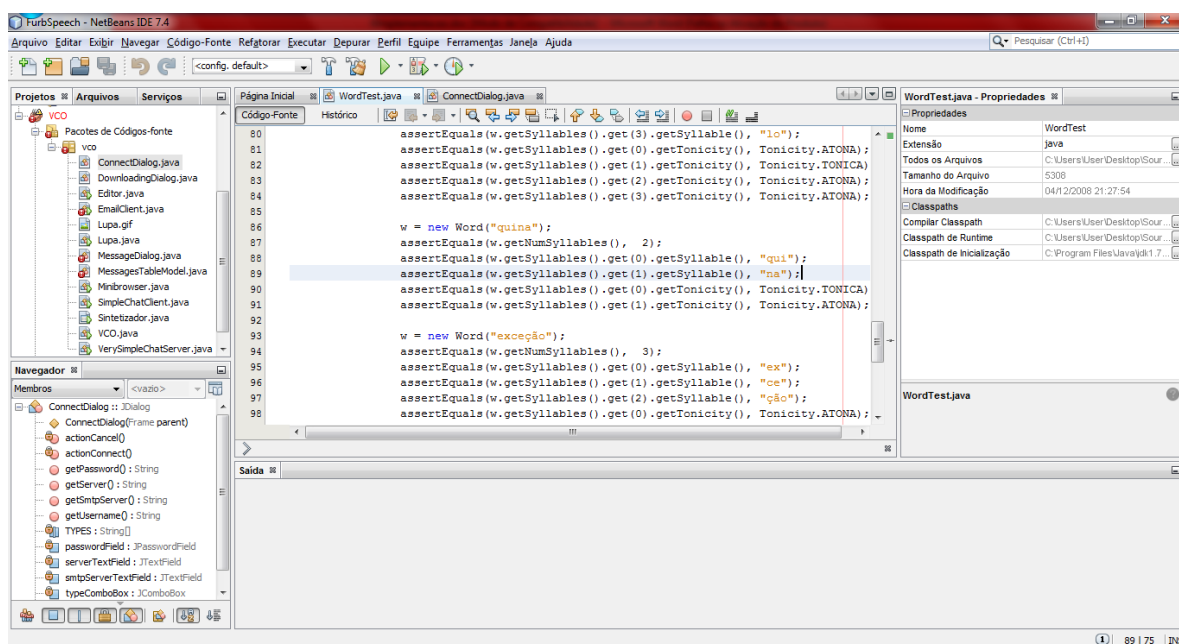



Figura 5.1: interface gráfica do IDE NetBeans.

5.2.3 MATLAB

O MATLAB, mostrado na Figura 5.2, é um ambiente de desenvolvimento integrado para o desenvolvimento de algoritmos e modelagem de sistemas, sendo considerado produto líder no mercado em cálculo numérico e de fácil uso.

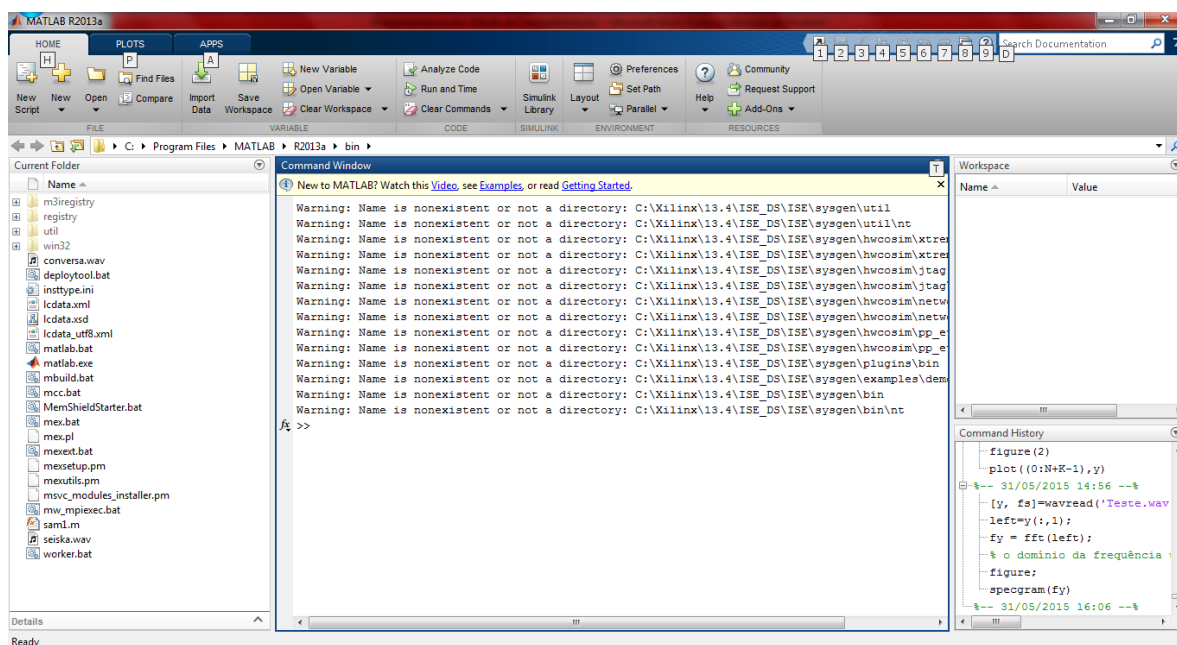



Figura 5.2: interface gráfica do *software* MATLAB.

O MATLAB possui funções de cálculo numérico, geração de gráficos, elaborador de interfaces gráficas denominado GUIDE, ambiente de modelagem e simulação de sistemas (SIMULINK) e *toolboxes* para desenvolvimento de simulações e aplicações científicas de naturezas diversas.

O MATLAB foi utilizado para análise e comparação da forma de onda no domínio da frequência do resultado gerado e uma voz natural gravada com um locutor real.

5.2.4 Editor de Áudio Audacity

O Audacity, mostrado na Figura 5.3, é um *software* para edição digital de áudio livre e gratuito disponível para as plataformas Microsoft Windows, GNU/Linux e Mac. O Audacity permite a manipulação de arquivos do tipo .WAV., .MP3 e OGG. Permite a gravação e reprodução de sons, além de apresentar recursos de edição simples como recortar, copiar, colar, apagar, mixagem em múltiplas faixas, aplicação de efeitos, remoção de ruídos, modificação de velocidade sem alterar a altura, nivelamento e equalização.

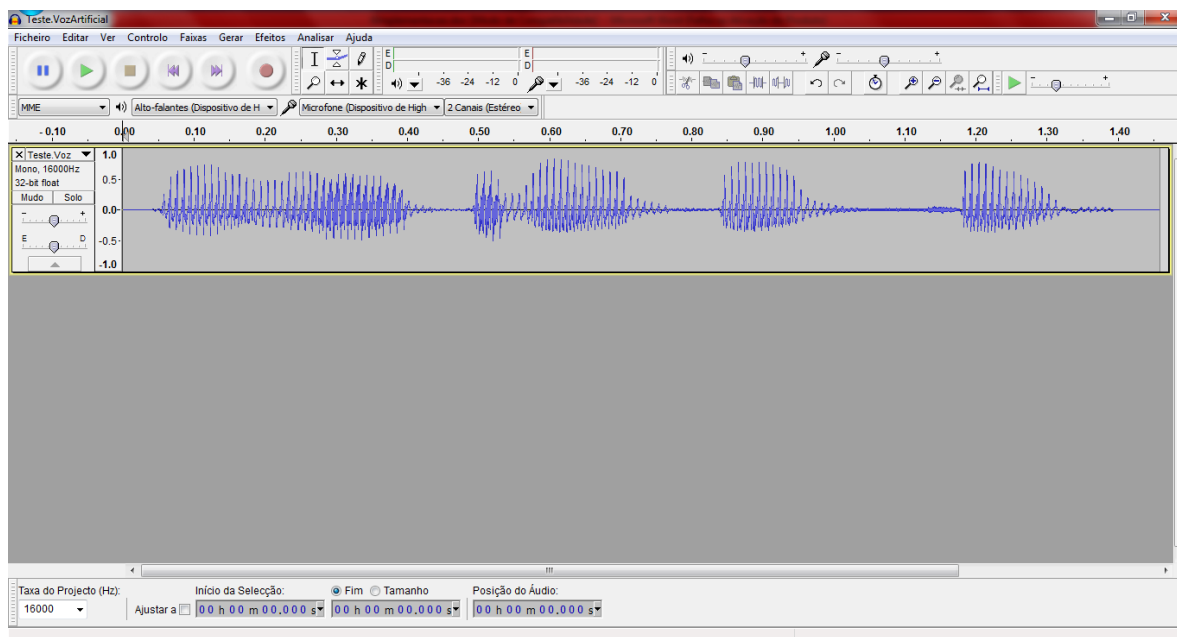



Figura 5.3: interface gráfica do editor de áudio Audacity.

O Audacity foi utilizado para análise e comparação da forma de onda no domínio do tempo do resultado gerado e uma voz natural gravada com um locutor real.

5.3 O sistema desenvolvido

5.3.1 Características gerais

Por apresentar resultados mais naturais e inteligíveis, além de sua simplicidade, baixo esforço computacional e ampla documentação científica disponível, a síntese concatenativa foi utilizada como método de síntese de voz, sendo o sintetizador escolhido o MBORLA.

O sistema desenvolvido trabalha em conjunto com o MBROLA, entretanto, o MBROLA, conforme dito anteriormente, não é um sistema TTS propriamente dito, pois não converte texto puro em fala, apenas aceita como entrada texto contendo dífonos e informações sobre a prosódia. Assim, a proposta apresentada por esta Dissertação é atuar como um *font-end* para o MBROLA, provendo para este último, as informações sobre dífonos e prosódia a partir de texto puro. Contudo, o sistema foi desenvolvido de tal forma que possa prover suporte para outras APIs, como Java Speech API e Google Translator API, e consequentemente suporte para outras línguas, com pouca alteração de código. Sendo necessário apenas uma linha de código para sintetizar uma frase.

Uma vez que o sistema foi desenvolvido com base na tecnologia Java, sua execução é possível em todas as plataformas que oferecem suporte a *Java Virtual Machine* e ao

MBROLA, como GNU/Linux e Microsoft Windows sem necessidade de recompilação, pois o sistema operacional é que se deve adaptar ao sistema, garantindo que a JVM e o MBROLA estejam instalados para a perfeita execução do *software*.

O sistema é totalmente baseado na filosofia de *software* livre, com código fonte aberto e de livre distribuição, para que a comunidade possa colaborar abertamente no desenvolvimento do projeto além de abrir possibilidade de personalização e modificação para aplicações específicas para eventuais interessados. Além disso, espera-se que as limitações apresentadas pelo sistema possam ser resolvidas ao longo do tempo em um prazo muito menor do que se o projeto proposto fosse proprietário. Ademais, o mesmo será gratuito, de forma a garantir aos deficientes visuais dos mais diferentes níveis sociais amplo acesso aos recursos oferecidos pela informática, gerando alto impacto na integração social desse grupo.

Quanto à síntese, observou-se apesar de que ainda robótico, procurou-se desenvolver um voz com tom grave e lento a fim de evitar o cansaço por exposição durante longos períodos de tempo e para garantir a clareza, respectivamente. O processo de síntese, apesar de pouco natural e apresentar complexidade computacional consideravelmente maior, foi escolhido em detrimento do uso de vozes pré-gravadas por este ocupar um considerável espaço em disco, o que poderia inviabilizar o "*port*" do projeto para plataformas embarcadas. Outrossim, vozes pré-gravadas são adequadas apenas quando já se conhece previamente o texto a ser falado, o que limita as possibilidades de interação com o usuário.

Com relação às etapas de síntese discutidas no Capítulo 3, coube ao *front-end* desenvolvido as quatro primeiras etapas (análise da estrutura, pré-processamento do texto, conversão de texto para dífonos e análise da prosódia), deixando para o MBROLA apenas a etapa de produção de forma de onda a partir dos dados providos pelo sistema desenvolvido (dífonos e dados de prosódia). A arquitetura do sistema proposto pode ser vista na Figura 5.4. Nesta Figura, os blocos em azul representam as entradas e saídas e o que foi efetivamente desenvolvido nesta Dissertação.

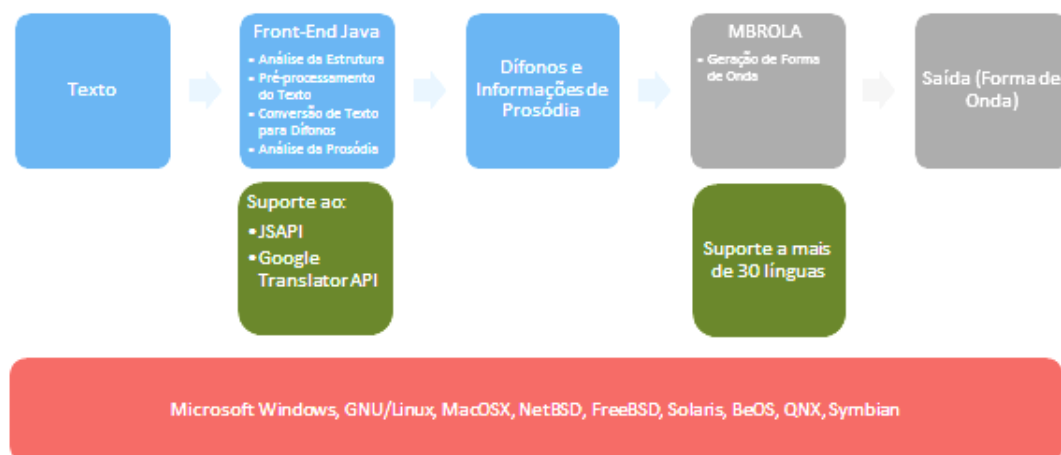



Figura 5.4: arquitetura proposta.

5.3.2 O front-end desenvolvido

O *front-end* tem por objetivo converter texto contendo símbolos, números e abreviações em sua forma por extenso, em um processo chamado de tokenização e, posteriormente, realizar a transcrição fonética e fornecer informações de prosódia a serem utilizados pelo *back end*.

O *front-end* possui algoritmos para normalização do texto baseado nas regras de conversão grafema-fonema, divisão silábica e marcação de sílaba tônica. De fora geral, os principais passos realizados pelo *front-end* desenvolvido são: Análise da Estrutura, Pré-Processamento do Texto, Conversão Texto-para-Fonema e Análise da Prosódia.

5.3.2.1 Entrada e Saída

O sistema recebe como entrada texto simples, sem elementos gráficos ou sinais de formatação de texto, e gera um arquivo .pho que informa ao MBROLA a lista de dífonos a serem concatenados e que contém os fonemas, conforme a representação mostrada na Tabela 5.1, com suas respectivas durações em milissegundos e curva de prosódia, esta última, por sua vez, é composta por um percentual indicador de posição, *pitch* - frequências fundamentais, e amplitude.

Em outras palavras, cada linha contém um fonema, a duração em milissegundos e a série de pitch do alvo composto por dois números em ponto flutuante: um representa a posição em um percentual da duração total e o valor seguinte representa o valor em Hertz do pitch na referida posição. Por exemplo, a linha:

_ 51 (25,114)

informa que o sintetizador deve produzir um silêncio de 51 ms com um pitch de 114 Hz a 25% desses 51 ms. As frequências fundamentais dos alvos definem a curva do pitch.

A curva de entonação é contínua, uma vez que o MBROLA realiza um decaimento automático da frequência ao se sintetizar fonemas não vozeados.

Os dados no arquivo são separados ou por espaços em branco ou por tabulações. Eventuais comentários podem ser inseridos nos arquivos .pho por meio de ponto-e-vírgula.

É importante frisar que o MBROLA é responsável por gerar o dífono, produzindo uma saída de áudio de 16 bits baseado no banco de dados br3 e que também pode ser redirecionada para um arquivo no formato .wav.

O *front-end* desenvolvido permite que seja definido tanto o local onde será salvo o arquivo .wav como qual será o banco de dados de dífonos e onde o mesmo se encontra.

5.3.2.2 Análise da estrutura

A análise da estrutura processa a entrada em texto a fim de determinar onde parágrafos, frases e outras estruturas começam e terminam. Dados sobre a pontuação e a formatação são usados nesse estágio, como vírgula, ponto e vírgula e ponto final e ponto parágrafo. Por se tratar de um protótipo, não foi agregado ao sistema um dicionário de abreviaturas e siglas.

5.3.2.3 Pré-processamento

O pré-processamento do texto analisa a entrada do texto buscando por construções especiais da linguagem, como acrônicos, abreviações, datas, horas, números, medidas, valores monetários, endereços de e-mails, entre outros.

O resultado dessas duas primeiras etapas é a forma falada do texto escrito, por exemplo:

R. Mário Mamede, 455, Bl. A, Ap. 203.”

“Rua Mário Mamede, número quatrocentos e cinquenta e cinco, bloco A, apartamento duzentos e três.”

“Depositar R$ 1.500,00 na minha conta”

“Depositar mil e quinhentos reais na minha conta”

“Nasci no dia 11/02/1988”

“Nasci no dia onze de fevereiro de mil novecentos e noventa e oito”

5.3.2.4 Análise morfossintática e linguística

A fim de eliminar qualquer ambiguidade na pronúncia das palavras, em especial de homógrafos não homófonos, é realizada a análise morfossintática por meio de um *parser* não determinístico.

O *parser* usa a sequência de entrada para determinar a estrutura gramatical conforme a gramática formal definida, transformando-a em uma árvore para processamento posterior e captura da hierarquia implícita na entrada.

O *parser* decompõe o texto em unidades estruturais a fim de que sejam organizadas dentro de um bloco de forma ascendente, *bottom-u*p, iniciando com a entrada de dados e reescrevendo-a até o símbolo inicial, tentando localizar os elementos mais básicos. Em conjunto com *tokens* e regras gramaticais, gera-se então a árvore sintática da estrutura de entrada.

Para os casos em que este é insuficiente, são necessárias e realizadas as análises semânticas e pragmáticas conforme o contexto.

5.3.2.5 Separação silábica e identificação das sílabas tônicas

Antes de iniciar a transcrição fonética, é realizada a separação silábica e a identificação das sílabas tônicas por meio da acentuação. Para a identificação de oxítonas não acentuadas podemos aplicar as seguintes regras já mencionadas conforme explicado no trabalho (AZUIRSON, 2009): palavras terminadas em "im" e "um"; palavras terminadas em "ar", "er", "ir" e "or", devido à forma infinitiva dos verbos apresentarem a sua última sílaba tônica; e palavras terminadas em z antecedidas por vogais. Vale ressaltar que embora essas regras não sejam verdadeiras para todos os casos, elas abrangem a maioria deles, apresentando um bom índice de acerto, conforme explicado em (AZUIRSON, 2009).

5.3.2.6 Conversão texto-para-fonema e transcrição fonética

Os passos restantes são os responsáveis por converter o texto falado em fala propriamente dita. A conversão texto-para-fonema, como o próprio nome sugere, converte cada palavra em fonema, lembrando que um fonema é a menor unidade sonora de uma língua. Diferentes línguas têm diferentes conjuntos de sons, ou seja, diferentes fonemas. Por exemplo, a língua inglesa apresenta aproximadamente 45 fonemas, incluindo sons de consoantes e vogais, enquanto que a língua japonesa apresenta menos fonemas e inclui sons não encontrados na língua inglesa.

Ao realizar a transcrição fonética, o sistema deve utilizar a mesma notação padronizada utilizada pelo MBROLA, de forma que sua saída seja uma entrada adequada e compatível com o MBROLA. Tal representação é mostrada na Tabela 5.1, que mostra a lista de fonemas seguidos por seus respectivos exemplos de ocorrência.

Tabela 5.1: representação dos fonemas utilizados para o MBROLA.

_	Silencio	i	Irmã	O	Opera	u	Utiliza
a	Ave	in	Indica	On	Onde	um	Umbigo
an	antigo	j	joaquim	Oo	Óculos	v	Valor
@	hão	k	Casa	P	Papa	w	wellington
b	baba	l	Luso	R	Real	x	Xarope
d	Dado	lh	Lhama	r2	Carta	y	ionosfera
e	Episcopal	m	mesmo	Rr	rapadura	z	Zebra
Ee	Era	m2	castram	S	Sapato	u	Utiliza
G	Gato	n	Nada	s2	Casca	um	umbigo
H	Habib	nh	nhoque	T	Taubaté	v	Valor

Durante a etapa de transcrição, é realizado um mapeamento por meio de *Look-up Tables* e árvores de decisão para a obtenção da representação fonética a partir do texto utilizando a representação fonética mostrada na Tabela 5.1, sendo aplicadas as regras de transcrição de fonemas estudadas nos Capítulos iniciais desta Dissertação. Nesta etapa também não foi implementado dicionário de exceções para pronúncia correta de palavras estrangeiras. Quando implementado, a busca no dicionário de exceções deve preceder a transcrição fonética. Caso a palavra não seja encontrada no dicionário, então se segue a divisão silábica, identificação da sílaba tônica e aplicação das regras de transcrição.

O sistema realiza a transcrição fonética conforme as regras explicadas no item 3.2.1.3 (Transcrição Fonética) da presente Dissertação. Além disso, aplicam-se as seguintes regras já mencionadas: As vogais apresentam som aberto quando acento é agudo e fechado quando circunflexo. A letra x, a mais problemática, é transcrita como /x/ em início de palavras, depois de "n" e depois de "ai", "ei" ou "ou"; como /z/ em palavras iniciadas com "ex" seguido de vogal; e /s/ quando seguido de consoante.

Para a seleção dos fonemas, foi utilizado um classificador, cuja tarefa é realizar o mapeamento dos atributos para classificação dos fonemas. No caso, o classificador adotado foi a árvore de decisão com base no algoritmo ID3. Foi utilizada a modelagem descritiva, um modelo de classificação é utilizado como ferramenta para distinguir diferentes fonemas de diferentes classes.

Uma árvore de decisão representa uma função discreta para representar dados a serem classificados. Uma árvore de decisão classifica as instâncias ordenando-as da raiz para algum nó-folha, onde cada nó da árvore representa uma classificação, sendo uma modelagem semelhante à regra "*if-then*".

Tal modelagem segue a estratégia "dividir para conquistar", em que um problema complexo é decomposto em subproblemas mais simples. A mesma estratégia é aplicada a cada subproblema, conforme mostrado no algoritmo abaixo em pseudocódigo:

```
Nó criaArvore(exemplos, alvo, atributos){
  se todos os exemplos tem mesmo valor de Alvo então
      retorna folha com valor;
  senão se o conjunto de atributos é vazio então
      retorna folha com o valor Alvo mais comum entre exemplos;
  senão{
    A <- melhor atributo com as variações v1,v2,v3...,vk;
    Particiona exemplos segundo valores para A em conjuntos S1, S2, ... Sk;
    Cria um nó de decisão N com atributo A;
    Cria nó de decisão N com atributo A;
    Para i:1 até K faça
      Conecta um nó B para o nó N com teste vi;
    Se si não é vazio então
      Conecta ramo B a criaArvore(si, alvo, atributos - {A});
    Senão então
      Conecta B para folha do nó com Alvo mais comum;
    Retorna N
  }
}
```

O algoritmo escolhe o melhor atributo para repartir as instâncias e criar o nó de decisão correspondente.

Árvores de decisão estão fundamentadas no paradigma *bottom-up* e seu uso se deve: ao fato dos fonemas serem classificados em termos de um conjunto de propriedades fixas (estudadas no item 2.5 do Capítulo 2); o número de classes é definido *a priori*; há uma quantidade bem maior de objetos do que classes; a tarefa de

classificação pode ser implementada de forma lógica, empregando uma base de regras de decisão, expressando a classificação de cada fonema como a descrição de uma expressão lógica.

Em uma árvore de decisão, o conhecimento é representado em cada nó que, ao ser testado, pode conduzir a busca a um de seus filhos. Deste modo, descendo da raiz em direção às folhas da árvore, pode-se selecionar a configuração do sistema, e deste modo comportamento associado.

A árvore de decisão implementada é de classe discreta, categórica não-ordinal (que assume um conjunto finito de valores que não podem ser ordenados).

O algoritmo heurístico mais conhecido para a escolha do melhor atributo é o ID3 e se baseia na escolha inicial de atributos que minimizem a entropia.

Se a informação é uma medida da quantidade de incerteza de um processo que ocorre com alguma probabilidade: $I(a_k) = -log_\alpha(p_k)$. Então a quantidade média de informação de uma fonte A é denominada entropia e esta é dada por: $H(A) = -\sum_{k=0}^{K-1} p_k log_\alpha(p_k)$.

O algoritmo continua até que uma das condições seja satisfeita: todos os atributos foram incluídos no caminho da raiz até as folhas ou os exemplos de treinamento associados com dado ramo apresentam o mesmo valor da saída.

O ID3 é um algoritmo pioneiro em indução de árvores de decisão, sendo um algoritmo recursivo de busca gulosa, procurando sobre um conjunto de atributos aqueles que melhor dividem os exemplos, gerando sub-árvores. A principal limitação do ID3 é que ele só lida com atributos categóricos não-ordinais, não sendo possível apresentar conjunto de dados com atributos contínuos, devendo, portanto, atributos contínuos serem discretizados previamente.

5.3.2.7 Entonação e prosódia

Por fim, a entonação é realizada por meio de sinais de ponto, exclamação e interrogação.

A análise da prosódia é responsável por processar a estrutura da sentença, palavras e fonemas para determinar a prosódia adequada. Conforme já dito, a prosódia inclui muitas das características da fala além dos sons produzidos, como melodia, ritmo, pausas, velocidade e ênfases. Uma prosódia apropriada é importante para uma produção de som mais natural.

A duração dos dífonos é baseada em valores estatísticos de acordo com o valor médio da distribuição dos valores que estes podem assumir acompanhados de uma

variação percentual (desvio-padrão) a fim de efetuar o aumento ou diminuição na duração do segmento. Além disso, tais valores devem ser maiores ou iguais a um determinado limiar. Eventuais ajustes empíricos foram realizados à medida que o sistema foi testado.

Além disso, a duração é influenciada pelo contexto fonético anterior e posterior, sendo limitada pelos segmentos vizinhos, ressaltando que palavras de conteúdo apresentam maior ênfase.

Por uma questão de limitação do MBROLA, os fonemas podem ser sintetizados com uma duração máxima que depende da frequência fundamental em que foram produzidas. Maior a frequência, menor a duração. Para uma frequência de 133 Hz, a duração máxima é de 7.5 s. Para a frequência de 66.5 Hz, a duração é de 15 s, e para a frequência de 266 Hz esse valor é de 3.75 s.

Assim, o conjunto composto pelo *front-end* e o MBROLA forma um sistema TTS completo, capaz de converter texto em sinal de voz.

É importante frisar que, embora o foco desta Dissertação seja a língua portuguesa, o uso do MBROLA permite que o sistema possa ser modificado para todas as línguas disponíveis para o mesmo, desde que se atente para as devidas modificações das regras de transcrição fonética e de prosódia.

O sistema não exige equipamentos robustos, muito menos *hardware* adicional.

5.3.4 O pacote de softwares desenvolvido

O projeto é composto por um sintetizador de voz, editor de texto, cliente de e-mails, chat, navegador web e lente de aumento, cujas respectivas interfaces são mostradas na Figura 5.5. A escolha a respeito das ferramentas presentes no pacote de *software* proposto deve-se ao fato de serem as aplicações mais comuns e úteis para usuários de computador em geral e, embora já existam algumas soluções acessíveis para aplicações como lente de aumento e sintetizador de voz, não há atualmente um pacote que integre todas essas aplicações, seja nativamente multiplataforma, livre e voltado para falantes da língua portuguesa. Além disso, as aplicações desenvolvidas visam validar o uso do sintetizador de voz com outras aplicações de forma a torna-las acessíveis.

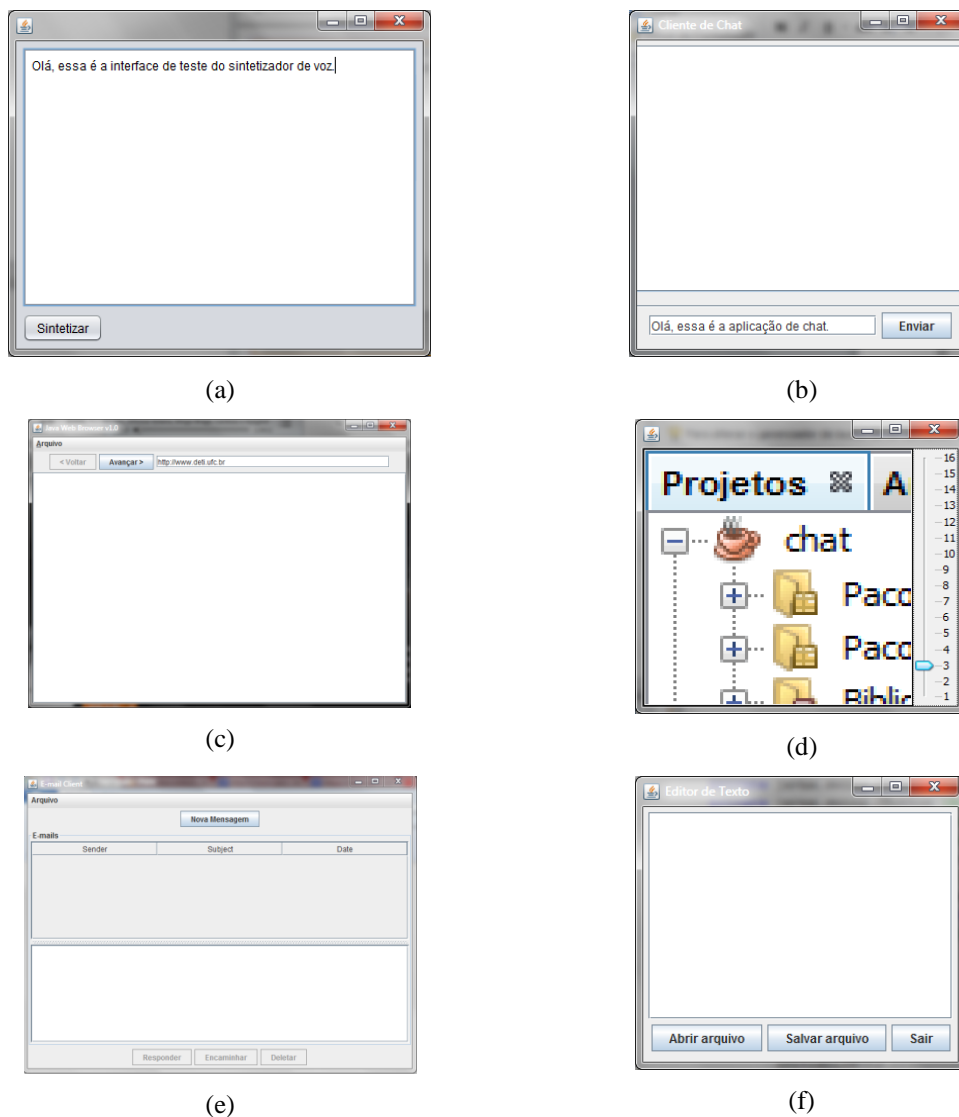



Figura 5.5: interface do (a) Sintetizador de Voz, (b) Aplicação de Chat, (c) Navegador de Internet, (d) Lente de Aumento, (e) Cliente de E-mail, (f) Editor de Texto.

O cliente de e-mail e o navegador web ainda se encontram em fase de desenvolvimento.

5.3.4.1 A lente de aumento virtual

A baixa visão corresponde a um comprometimento importante da função visual, porém não equivale à cegueira. Baixa visão e/ou visão subnormal são termos usualmente empregados para definir a situação em que o olho está com uma de suas vias de condução do impulso visual alterada de maneira irreversível, cuja perda visual constitui um obstáculo para o desenvolvimento normal da vida do indivíduo e que precisa de correção especial.

Uma das maiores dificuldades enfrentadas no desenvolvimento de *softwares* voltados para esse grupo está no tratamento de imagens que propicie aos usuários uma apresentação confortável e, tanto quanto possível, isenta de distorções. Uma técnica bastante usada nesses casos é a chamada operação de reamostragem, também conhecida por *zoom in*.

Trata-se de uma operação que consiste, basicamente em adicionar linhas e colunas vazias na imagem original, aumentando a sua resolução espacial. Cores, então, são atribuídas a estas linhas e colunas utilizando-se um dos seguintes métodos: replicação do vizinho mais próximo; interpolação linear; interpolação bi-linear ou interpolação bicúbica.

Alguns trabalhos apresentam os seguintes resultados de um *zoom-in* a partir da replicação do vizinho mais próximo: para um fator de ampliação de 2x, o resultado do encontrado é satisfatório. Entretanto, para fatores maiores, surgem blocos de cores homogêneas na imagem. Melhores resultados seriam obtidos por meio de outros algoritmos, como usar o filtro espacial de Bartlett (interpolação bilinear) para implementar o *zoom-in*. Os métodos de interpolação em imagens atuam como filtros passa-baixa, atenuando as altas-frequências nas imagens resultantes, causando um efeito de sombreamento na imagem (BIDARRA 2005).

Entretanto, pelo fato dos usuários serem pessoas com baixa visão, é necessário resgatar as altas-frequências na imagem ampliada, caracterizadas pelas regiões de borda presentes na imagem. Para tanto, filtros de realce ou detecção das bordas presentes na imagem digitalmente ampliada, tornam-se imprescindíveis.

Um outro problema que aparece nesse tipo de processamento é o serrilhado (*aliasing*), no momento seguinte à ampliação da imagem. A solução desse problema, porém, é crítica. Seu custo computacional é alto e a aplicação em questão é dependente de respostas em tempo real. Assim, a pesquisa e/ou desenvolvimento de um algoritmo de *anti-aliasing* eficiente torna-se igualmente necessário.

Não basta aos ampliadores um tratamento baseado apenas no tamanho da letra, na cor de fundo e/ou no contraste da tela. Há mais parâmetros em jogo nesse processo: profundidade, luminosidade, contorno, tanto da letra quanto do desenho exibido.

Seguindo a necessidade de desenvolver facilmente uma interface gráfica simples - o que implica programação gráfica, em um sistema que tem por objetivo ser portável, a lente de aumento virtual foi desenvolvida também usando a linguagem Java e fazendo

uso da replicação do vizinho mais próximo por uma questão de simplicidade e velocidade.

5.3.4.2 Editor de Texto

O editor de texto desenvolvido apresenta uma interface simples e com poucas funcionalidades, como abrir, salvar, sair, copiar, recortar e colar, semelhante ao Gedit do Gnome ou Notepad do Microsoft Windows, trabalhando com textos simples, sem imagens ou itens de formatação, como cor, estilo do texto, etc. Ao salvar o arquivo, o sistema lê o texto que foi digitado.

5.3.4.3 Aplicações de *Chat*

Foi desenvolvido também um cliente e um servidor de chat bastante simples baseado em socket Java, sendo necessário informar o IP da máquina com a qual se deseja conectar. Ao pressionar a tecla “Enter”, o cliente “lê” e envia a mensagem digitada ao destinatário. Quando a mensagem chega ao último, o sistema “lê” a mensagem para o destinatário.

6. TESTES, RESULTADOS OBTIDOS E DISCUSSÕES

Esta Capítulo visa apresentar e discutir a metodologia dos testes realizados com a ferramenta desenvolvida bem como os seus resultados.

Certamente o fator que mais pesa na aceitação por parte dos usuários de sintetizadores de voz é a qualidade na saída resultante. Saber como avaliar a qualidade da síntese e os fatores que influenciam nesta são ponderações muito importantes no processo de desenvolvimento de *softwares* de acessibilidade. A qualidade de um sistema de síntese de voz é julgada de acordo com sua similaridade com a voz humana.

Um sistema de síntese de voz é comumente avaliado sob três aspectos: precisão no tratamento do texto de entrada, inteligibilidade – o percentual do resultado que foi corretamente entendido; e naturalidade - o quão parecido é a saída com uma voz humana real do resultado (SCHROETER,2005).

Por precisão entende-se a habilidade de ler uma entrada de texto da mesma forma que um ser humano leria, estando relacionado com o funcionamento correto do *front-end*. Projetistas de sintetizadores baseados em formantes devem se concentrar em maximizar inteligibilidade, aceitando o fato de que a naturalidade é difícil de ser atingida. É comum sistemas concatenativos enfatizarem em excesso a naturalidade, negligenciando a inteligibilidade. A precisão pode ser avaliada verificando a correta síntese de abreviações e acrônicos e julgando o texto de saída gerado pelo *front-end*. Avaliar inteligibilidade e naturalidade requer testes de audição mais elaborados (SCHROETER, 2005).

Avaliação de um sistema TTS exige testes subjetivos. A União Internacional de Telecomunicações recomenda metodologias específicas de testes. Tais testes envolvem geralmente cinco pontos dentro de uma escala geral em critérios como "impressão geral", "esforço para compreensão", "compreensão", etc. Alternativamente, pode-se propor que voluntários expressem sua preferência dentre dois sistemas acerca de qual dos dois é melhor, testes A/B (SHAUGHNESSY, 2003; SCHROETER, 2005).

Manter um dicionário de pronuncia de itens específicos é uma solução interessante quando se pretende usar o sistema para determinadas aplicações.

Escolhas de engenharia típicas como *trade off* entre velocidade e memória, qualidade e complexidade, tempo de desenvolvimento e pressão do mercado são também frequentes no desenvolvimento de *softwares* de síntese de voz (SCHROETER, 2005).

6.1 Comparação com outros sintetizadores de voz

É importante lembrar que não foi encontrado durante a pesquisa um pacote de *softwares* acessíveis, livre, gratuito, de código aberto, nativamente multiplataforma disponível para falantes do português-brasileiro, contendo aplicações mais comuns no dia-a-dia de um usuário integradas a um sistema de síntese de voz.

Por fazer uso da tecnologia Java, a arquitetura proposta é nativamente multiplataforma - ao contrário do DOSVOX, que foi desenvolvido nativamente para o Windows e que é executado no GNU/Linux apenas se houver o Wine instalado, ou do ADRIANE, que é um ambiente puramente GNU/Linux. Embora soluções baseadas em plataformas livres sejam as ideais, tanto por ter uma filosofia de desenvolvimento colaborativo - e consequentemente mais rápido, como pelo baixo custo, não se pode forçar os usuários a adotarem um sistema operacional com o qual os usuários possivelmente não estejam habituados a usar.

Aplicações como JAWS e Virtual Vision custam aproximadamente US$ 1.200,00 e US$ 2.500,00, respectivamente, o que os torna inviáveis para usuários com condições financeiras restritas. Além disso, alguns dos sistemas que foram apresentados anteriormente apresentam síntese sofrível para o idioma português brasileiro e não fornecem a seus usuários ferramentas acessíveis integradas.

6.2 Resultados da síntese: análise quantitativa

Os testes iniciais tiveram por objetivo analisar, no domínio do tempo e da frequência, as diferenças entre a voz sintetizada e a voz natural, de forma a esclarecer quais parâmetros matemáticos influenciam na qualidade do resultado produzido a fim de que, posteriormente, possam ser realizados estudos com o intuito de melhorar a qualidade da síntese.

A forma de onda da frase “Olá, professor!” no domínio do tempo é mostrada na Figura 6.1a. O arquivo gerado, no formato “.wav” e de tamanho 45.6Kb apresenta taxa de amostragem de 256 kbps. O resultado foi obtido com auxílio do *software* Audacity [v. 2.0.5].

O resultado foi comparado com uma gravação da mesma frase realizada por um locutor humano em ambiente livre de ruído, sendo a forma de onda no domínio do tempo mostrado na Figura 6.1b.

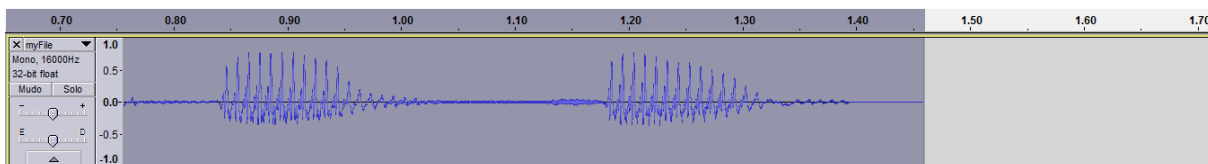



Figura 6.1a: resultado da forma de onda no domínio do tempo para a frase "Olá, professor" gerada pelo sintetizador.

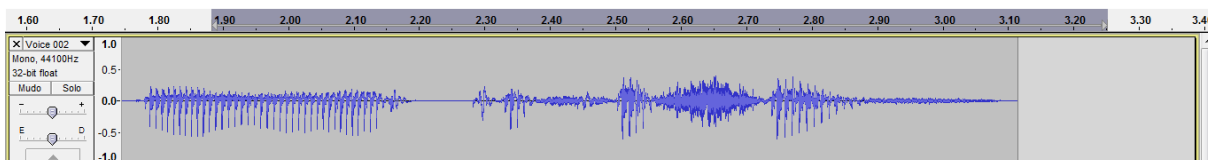



Figura 6.1b: resultado da forma de onda no domínio do tempo para a frase "Olá, professor" gerada por locutor humano.

Por meio do mesmo *software*, foi possível obter o espectro no domínio da frequência em dB x Hz, usando janela de Hanning, mostrado na Figura 6.2a.

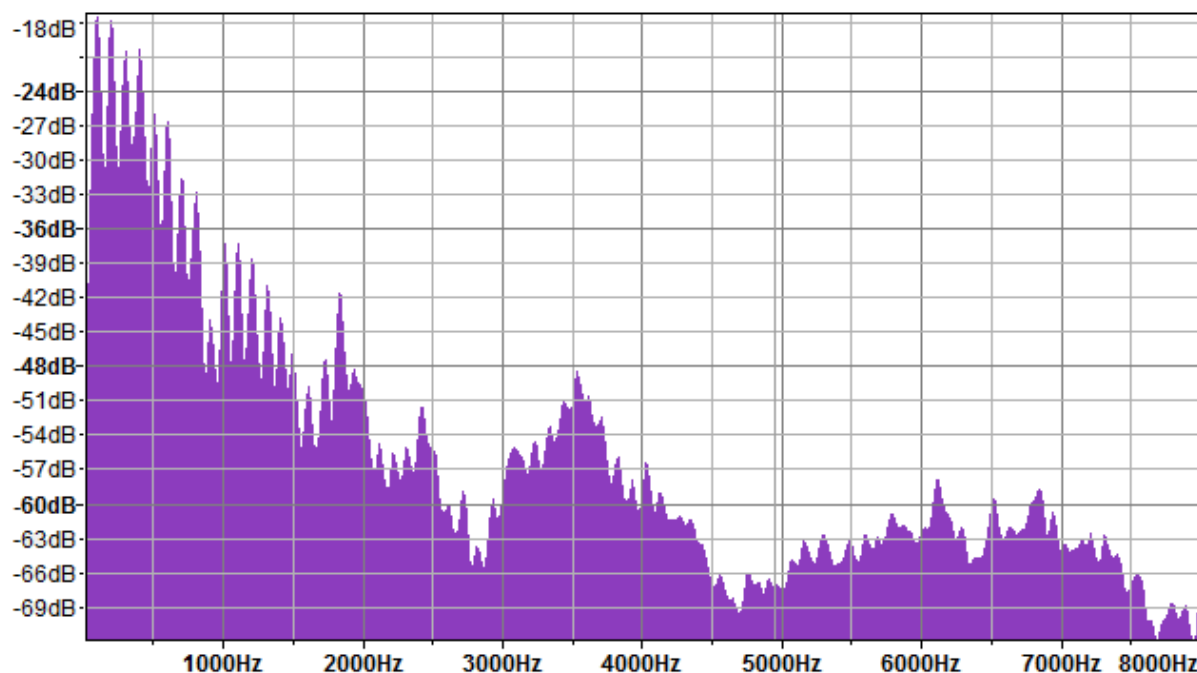



Figura 6.2a: resultado da forma de onda no domínio da frequência para a frase "Olá, professor" gerada pelo sintetizador.

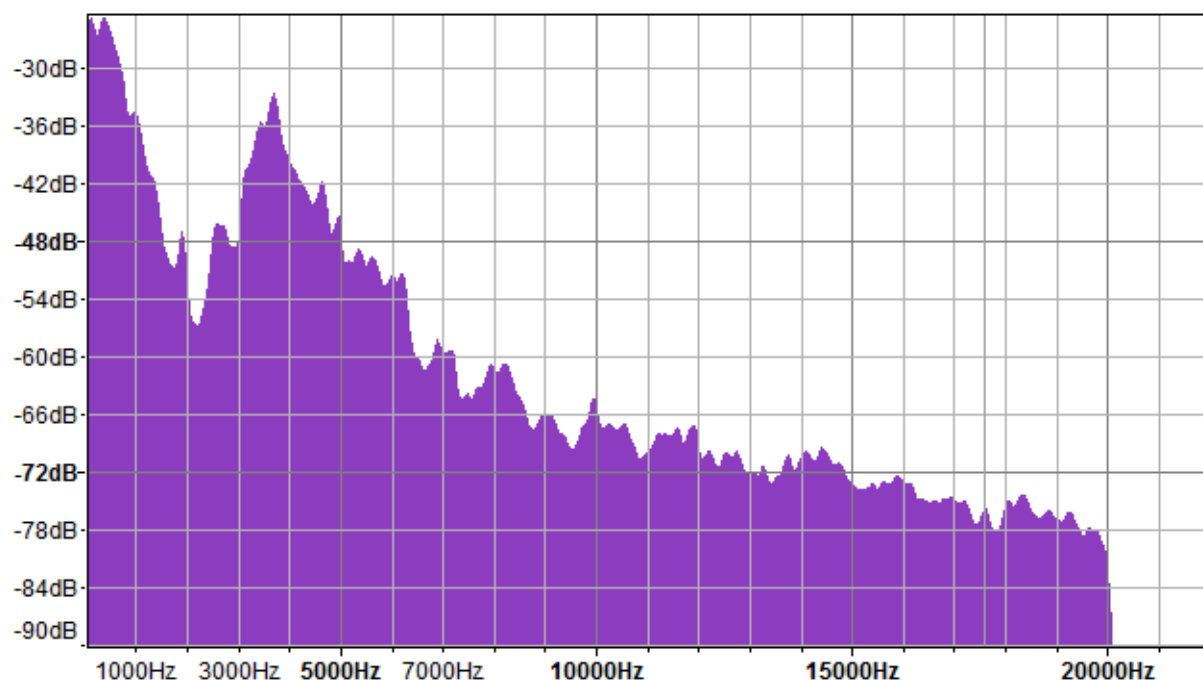



Figura 6.2b: resultado da forma de onda no domínio da frequência para a frase “Olá, professor” gerada por locutor humano.

Um espectrograma, ou sonograma, é a representação da variação tempo-frequência em que o valor em um dado ponto, isto é, a potência de uma dada frequência num dado instante de tempo é representado por um nível de uma cor em uma dada escala cromática. Por fim, utilizando o comando specgram, do Matlab v. 2013a, foi gerado o espectrograma, gráfico Frequência x Tempo, mostrado na Figura 6.3a, por meio dos seguintes comando:

```
[y, fs]=wavread('Teste.wav');          % lê o arquivo de áudio
left=y(:,1);
fy = fft(left);                        % transforma forma de onda do domínio do tempo
para
                                       % o domínio da frequência usando FFT
figure;
specgram(fy)                           % exibe espectrograma
```

Sendo o mesmo resultado comparado com o espectrograma da voz natural, mostrado Figura 6.3b.

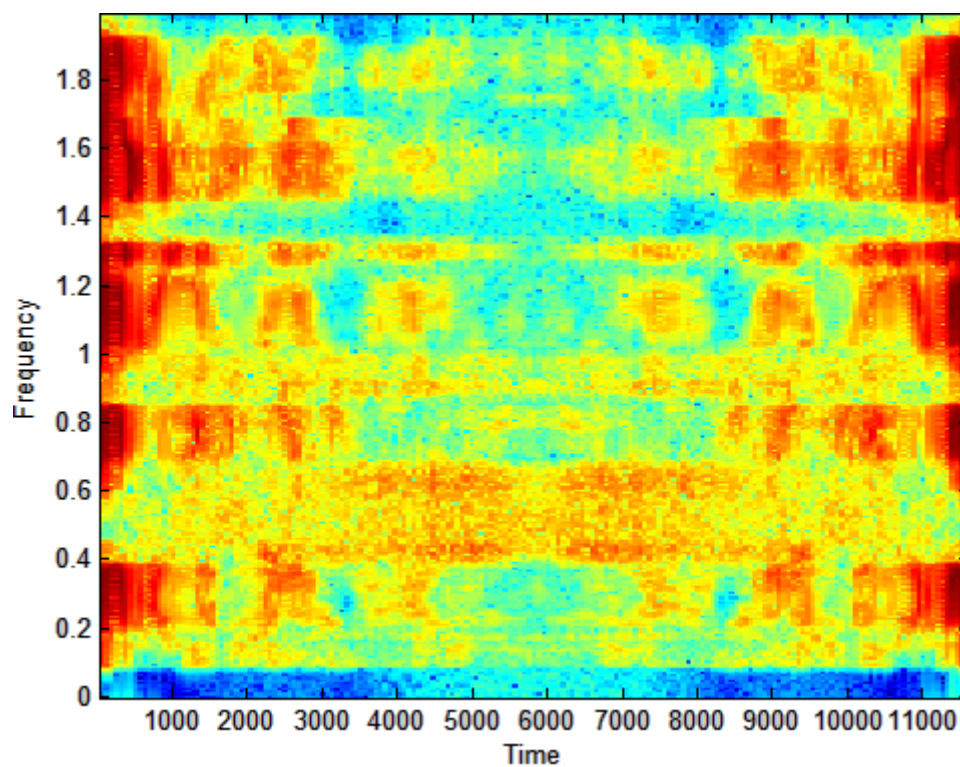



Figura 6.3a: espectrograma obtido para a frase “Olá, professor” gerada pelo sintetizador.

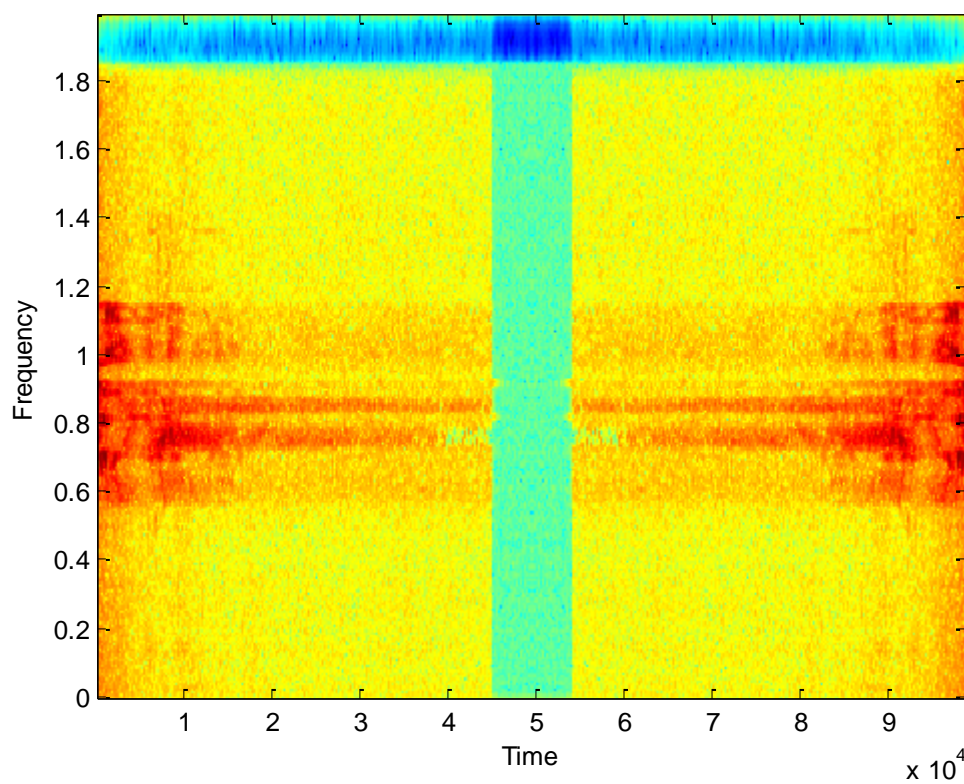



Figura 6.3b: espectrograma obtido para a frase “Olá, professor” gerada por locutor humano.

O que se pode observar nestes resultados é a ausência dos pontos de concatenação na voz natural. Fato facilmente observado no espectrograma mais “suave” da voz natural que mostra a variação gradual de potência (representado pela variação mais contínua da cor), em comparação com o resultado descontínuo gerado pelo sintetizador.

Percebe-se também que a voz natural apresenta muito mais conteúdo em termos de informação, pois sua forma de onda é mais “cheia”. Isto é resultado da modelagem da voz, que não considera todos os parâmetros para a produção de voz, e da compressão realizada para armazenamento de dífonos no banco de dados. Tal fato é confirmado pela análise do espectro no domínio da frequência, no qual se percebe a maior presença de harmônicos na voz natural pela largura do espectro, tanto na intensidade como na largura do espectro.

6.3 Testes em campo: análise qualitativa

O estudo de usabilidade garante que o usuário consiga completar tarefas básicas. Tal estudo exige uma versão preliminar do *software* - a função principal do sistema deve estar completamente implementada. O teste deve ser conduzido em um laboratório e em um ambiente semelhante àquele em que o usuário final deve usar o sistema. Um questionário pode ser usado a fim de coletar sugestões, comentários e opiniões.

Com base nisso, foram realizados diversos testes tendo como usuários portadores de deficiência visual. Um dos testes contou com participação de uma deficiente visual de 40 anos, usuária de *softwares* de acessibilidade desde 1994, usando atualmente o DOSVOX. O teste foi realizado na Secretaria de Acessibilidade da Universidade Federal do Ceará. Foram comparados incialmente três sintetizadores de voz: o primeiro baseado em uma API da Google, o segundo baseado no FreeTTS e o último a implementação proposta pela presente Dissertação (nesta sequência), de tal forma que a usuária não tinha consciência de qual sintetizador estava sendo testado no momento. O teste em campo seguiu três etapas, descritas no questionário exibido no “Anexo B”, descritos a seguir.

(1) Naturalidade da fala: foi questionado à voluntária se a voz a qual escutava era um áudio pré-gravado ou se era voz sintetizada por computador. Além disso, foi solicitado que, em uma escala de 1 (muito ruim) a 5 (excelente), fornecesse uma pontuação sobre a qualidade da voz.

(2) Teste de Inteligibilidade: Foi solicitado que a usuária escutasse duas frases: "Olá, seja bem vinda ao projeto LESC Vox. Obrigada por usar o nosso sistema." e "Seja bem-vindo ao projeto de acessibilidade “Ver com os ouvidos”! O que você gostaria de fazer?". Pediu-se então que a usuária enumerasse quantas palavras não conseguiu entender ou entendeu errado (após ser informada o que de fato tinha sido "falado"), de tal forma que a usuária não tinha conhecimento prévio do que viria escutar.

(3) Teste de Usabilidade: Foi solicitado que a usuária usasse o sistema de forma independente para abrir aplicações específicas. Ao inicializar o sistema, a usuária deveria ser capaz de abrir as aplicações como editor de texto, cliente de *chat*, ou qualquer outra aplicação de sua vontade e utilizar sem necessidade de auxílio.

Por fim, foi solicitado que a usuária tecesse comentários gerais e sugerisse melhorias no sistema. A seguir é descrito o comparativo entre os três sistemas:

A. Sistema baseado no Google API: a usuária classificou o som como sintetizado e atribuiu conceito "razoável" à qualidade do som. Considerou som muito agudo, sugerindo alterar o tom e a velocidade. Informou ainda que tons muito agudos são desagradáveis quando escutados por muito tempo. Informou ainda que apresentou entonação errada por vezes mas em nada afetou a compreensão, apresentando 100% de entendimento.

B. Sistema baseado no FreeTTS: a usuária classificou o som como sintetizado e atribuiu conceito "muito bom" quanto à qualidade da síntese. Entretanto, notou que o sistema faz uso da fonética do inglês e, embora as frases fossem em português, isto tornou a escuta ininteligível. A usuária acredita que se houvesse modificação da fonética, apresentaria índice de inteligibilidade considerável, mas o sistema como foi apresentado recebeu conceito "muito ruim". Informou ainda que acredita que o sistema se mostrava bastante adequado para língua inglesa.

C. Sistema proposto: a usuária classificou o som como sintetizado e atribuiu conceito "muito bom", entendendo as frases em sua totalidade. Afirmou que o sistema apresenta tonalidade grave muito próximo do que julga ideal. Sugeriu apenas que fosse fornecida ao usuário uma forma de modificar a velocidade e o tom de voz.

Como comentários gerais, afirmou que não usaria de forma alguma o sistema baseado no FreeTTS (proposta B), ficando com as propostas A e C, informando ainda que as opções A e C apresentam síntese quase humanas, "não deixando a desejar de jeito nenhum" (*sic*) quando comparado com todas as ferramentas as quais teve acesso, como, por exemplo, DOSVOX e NVDA, e disse que o projeto se encontrava "no rumo certo" (*sic*).

O sistema foi testado tanto em ambientes GNU/Linux como em Microsoft Windows, apresentando em ambos a mesma qualidade.

Além de servir de *front-end* para o MBROLA, o sistema pode ser modificado facilmente para prover suporte ao Java Speech API e API do *Google Translator*, fornecendo suporte para diversas línguas estrangeiras além do português.

6.4 Testes em campo: análise quantitativa

Por se tratar de um critério subjetivo do ouvinte, avaliar vozes e falar humanas é uma tarefa difícil de ser realizada (COSTA e MONTE, 2012). Neste trabalho, foram usadas como principais métricas o MOS e o WER/WAR.

O MOS (*Mean Opinion Score*) é uma métrica se baseia na média de conceitos que vão de 1 a 5, obedecendo a seguinte escala:

1 - Muito ruim;

2 – Ruim;

3 – Razoável;

4 – Bom;

5 – Excelente.

Assim, o MOS é utilizado para verificar: a naturalidade e a inteligibilidade da fala. No que diz respeito à naturalidade da fala, o ouvinte é convidado a identificar se uma determinada fala que ouviu é natural e tentar distinguir se foi produzida artificialmente ou por um ser humano. Caso identifique ser artificial, pergunta-se o quão perto do natural a fala sintetizada se aproxima. Quanto à inteligibilidade da fala, o ouvinte é convidado a ouvir uma frase, devendo-se então verificar se o mesmo compreendeu o que foi dito, se a mensagem foi clara o suficiente e o quão difícil ou não foi a compreensão

Um teste MOS geralmente envolve de 12 a 24 usuários (SPANIAS, 1994). Juntamente com o MOS, outras duas métricas complementares entre si, são usadas em testes para assegurar a qualidade de plataformas de síntese de voz: o WAR (*Word*

Accuracy Rate) e o WER (*Word Error Rate*). O ouvinte deve expressar quantas palavras (não) consegue entender, acertou (ou errou) ou apresentou grande facilidade (dificuldade) para entender, podendo ser expresso em porcentagem do total da frase o número de palavras que (não) compreendeu.

O WER representa o número de palavras não entendidas em relação ao total de palavras em termos percentuais. O WAR representa o número total de palavras entendidas em relação ao total de palavras da frase, assim: $WER + WAR = 100\%$ (COSTA e MONTE, 2012).

Além do MOS, podem ser usadas as métricas DAM (*Diagnostic Acceptrability Measure*) e o DRT (*Diagnostic Rhyme Test*). O DRT é um teste de inteligibilidade cuja tarefa é reconhecer uma de duas palavras dentre o conjunto de pares com sons semelhantes.

Assim, para o presente trabalho, realizou-se uma bateria de testes MOS e WAR, envolvendo 20 voluntários videntes de ambos os sexos com idade entre 17 e 31 anos no Centro de Humanidades da Universidade Federal do Ceará, cujos resultados são mostrados na Tabela 6.1.

Tabela 6.1: valores MOS e WAR.

Voluntário	Idade	Sexo	MOS	WAR
Voluntário 1	30	F	3	100%
Voluntário 2	29	M	2	100%
Voluntário 3	26	M	2	50%
Voluntário 4	26	F	3	90%
Voluntário 5	20	F	4	70%
Voluntário 6	21	F	4	90%
Voluntário 7	20	M	3	40%
Voluntário 8	18	M	2	100%
Voluntário 9	18	M	3	90%
Voluntário 10	18	M	4	80%
Voluntário 11	19	M	4	70%
Voluntário 12	28	M	3	90%
Voluntário 13	20	M	3	50%
Voluntário 14	17	M	3	80%

Tabela 6.1: valores MOS e WAR (Continuação).

Voluntário 15	22	F	3	70%
Voluntário 16	20	F	3	75%
Voluntário 17	26	F	3	100%
Voluntário 18	30	F	3	100%
Voluntário 19	19	F	3	40%
Voluntário 20	18	F	3	100%
Valor Médio	22,25	-	3,05	79%

Fonte: Próprio autor.

Os gráficos para o MOS e WAR são mostrados respectivamente nas Figuras 6.4 e 6.5.

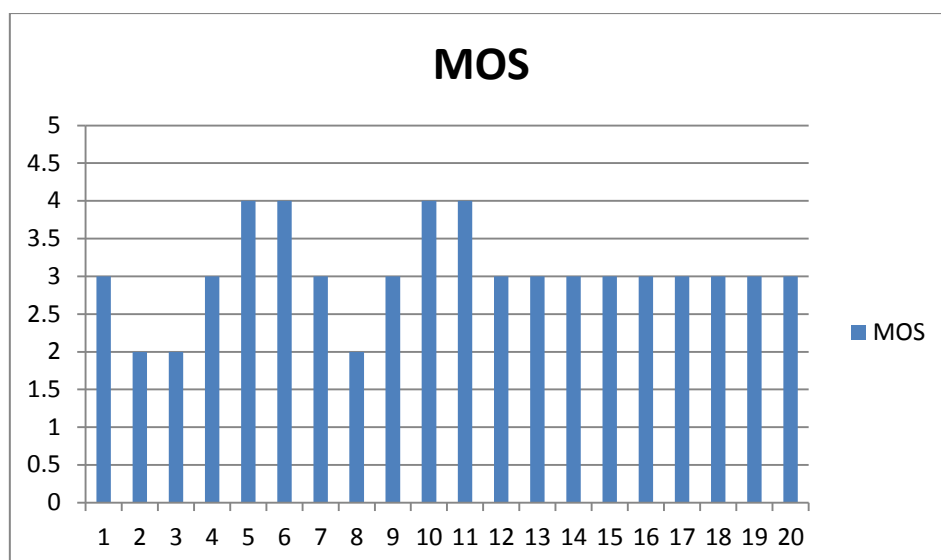



Figura 6.4: resultados para o MOS.

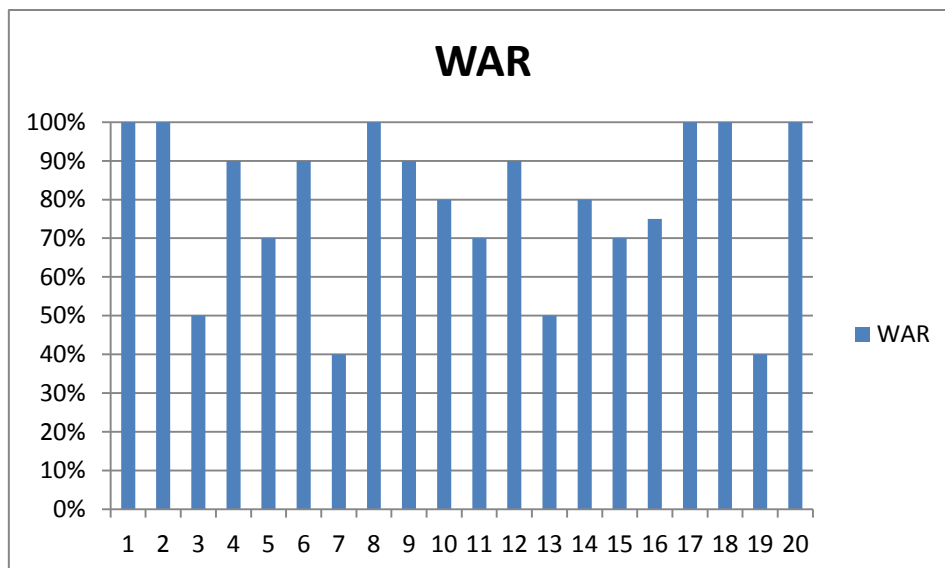



Figura 6.5: resultados para o WAR.

O resultado apresentou desvio-padrão de 0,60 para o MOS e 0,20 pra o WAR. A principal reclamação por parte dos voluntários está na descontinuidade inerente à técnica de concatenação.

Pode-se afirmar que o sistema proposto, embora ainda não tão natural quanto se deseje, apresenta boa inteligibilidade. Tal conclusão tem como base a comparação entre a voz produzida artificialmente por meio do sintetizador proposto e uma voz produzida por um locutor, natural, tanto no domínio do tempo como da frequência, considerando-se depoimento de usuária portadora de deficiência visual e o resultado dos testes de usabilidade bem como os resultados dos testes MOS e WER com voluntários,

O resultado mais inesperado residiu no depoimento da usuária deficiente visual: embora o resultado obtido com a API do *Google Translator* fosse mais natural, o tom grave do resultado obtido neste trabalho mostrou-se mais agradável, confortável e adequado para uso a longo prazo, um resultado não muito comentado em trabalhos envolvendo acessibilidade.

Embora haja vários projetos com características semelhantes, a flexibilidade do projeto, tanto pelo fato de atuar como *front end* para diversas APIs de síntese de voz, prover suporte para diversas línguas e diversas plataformas operacionais e ser livre, bem como o fato de já fornecer ao usuário um pacote de *softwares* mais utilizados, torna a solução proposta uma contribuição importante ao processo de integração digital de deficientes visuais.

7. CONCLUSÃO

Programadores e engenheiros de *software* envolvidos com *softwares* de acessibilidade devem considerar a qualidade e a naturalidade da síntese realizada. Um projeto de *software* acessível tem requisitos diferentes dos *softwares* convencionais e deve ser adotada uma abordagem específica desde o início das especificações do projeto, assim, *softwares* que visem atingir também o público com restrições visuais, devem ter esta meta estabelecida desde os requisitos iniciais do sistema. Considerar esses aspectos é um passo crítico para melhorar a qualidade de vida de usuários deficientes visuais.

A solução mostrada neste trabalho visa preencher uma lacuna existente nos *softwares* de acessibilidade com suporte à língua portuguesa, apresentando não só um sintetizador de voz com qualidade acima da média, como também apresenta um pacote de *softwares* pronto para uso e simples de usar. Os usuários atingiram os objetivos propostos e completaram as tarefas sem maiores dificuldades. As sugestões dadas pelos voluntários devem ser implementadas em versões futuras do *software*.

Embora não propriamente original, o projeto vem a atender uma demanda quase sempre ignorada pela indústria de TI, além disso não foi encontrado na literatura pesquisada um FRONT-END que realizasse mapeamento fonema-dífonos para a língua portuguesa. O fato de ser gratuito permite que pessoas pertencentes a qualquer classe social possam se beneficiar dos recursos oferecidos pela informática e o modelo aberto e colaborativo possibilita o rápido aprimoramento das ferramentas oferecidas pelo pacote desenvolvido, ao contrário do que costuma acontecer com sistemas fechados. O fato do sistema ser multiplataforma garante que usuários dos principais sistemas operacionais possam fazer uso dos benefícios pelo sistema, não forçando o usuário a adotar um sistema operacional com o qual esteja pouco habituado muito menos um que seja proprietário.

Os testes realizados abordaram tanto aspectos quantitativos como qualitativos, e em ambos, provou-se que, embora ainda haja trabalho a ser feito no tocante a tornar a voz mais natural, devendo ainda eliminar descontinuidades, o resultado é bastante inteligível e causa menos cansaço aos usuários que determinados outros sistemas com síntese mais semelhante à voz humana.

Diante do exposto, pode-se afirmar que o projeto proposto neste trabalho se apresenta como uma solução viável como forma de integrar socialmente deficientes

visuais e contribuindo para a diminuição da sua exclusão digital, quebrando barreiras e assegurando assim, um direito que é garantido pela constituição: o acesso livre a informação para todos os brasileiros de forma igualitária.

7.1 Trabalhos futuros

Dentre os trabalhos futuros que podem dar continuidade ao presente trabalho, podem-se citar: melhorias na qualidade da síntese de voz, incluindo prosódia e melhor reconhecimento de contexto para valores numéricos e abreviações, por meio de um dicionário, inserindo também meios de modificar velocidade e tom de voz por parte do usuário. Como forma de melhorar a qualidade da voz sintetizada, propõe-se implementar a solução proposta em (KANG et. al., 2009) para melhorar a coarticulação; conclusão da implementação de algumas ferramentas, como navegador web, sistema de Voz sobre IP e agenda; portar o sistema para plataformas móveis baseadas no sistema Android.

A síntese de dífonos tem apresentado resultados superiores em dispositivos móveis quando comparados com outras técnicas de síntese de voz. (TALAFOVÁ et. al., 2007) apresenta uma primeira aplicação de síntese de dífonos em ambiente móvel, cujo diagrama de funcionamento é mostrado na Figura 7.1. Neste trabalho, ao receber uma mensagem SMS, por exemplo, o sistema concatena amostras de voz pré-gravadas e armazenadas em um banco de dados.

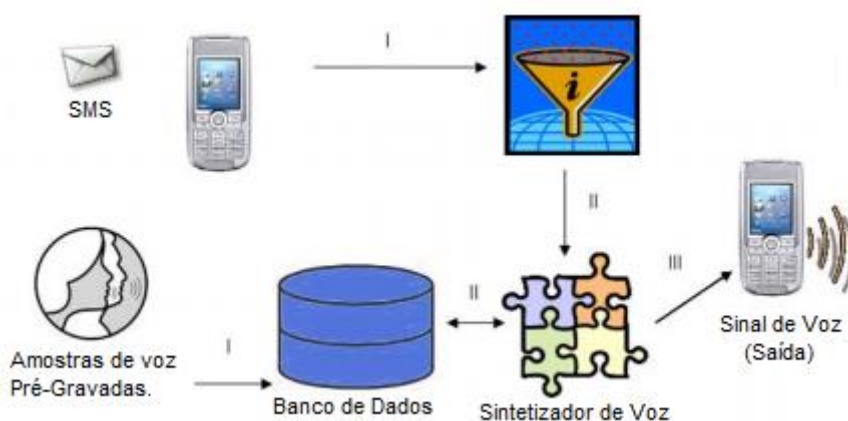



Figura 7.1: solução proposta em (TALAFOVÁ et. al., 2007) para aplicação em dispositivos móveis. Fonte: (TALAFOVÁ et. al., 2007 - Traduzido).

REFERÊNCIAS

ACAPELA - SITE OFICIAL. Disponível em: <http://www.acapela-group.com/acapela-for-linux-embedded/>. Acesso em: Outubro 2014.

AZUIRSON, Gabriel de Albuquerque Veloso. **Investigação da modelagem linguística e prosódica em sistemas de síntese de voz**. Trabalho de Conclusão de Curso em Engenharia de Computação. 64p. Universidade Federal de Pernambuco. Recife, 2009.

BIDARRA, Jorge; DIÓGENES, Carlos Eduardo Rodrigues. **XLUPA - Uma lente de aumento digital inteligente para pessoas com baixa visão**. In: III Seminário e II Oficina "Acessibilidade, TI e Inclusão Digital". São Paulo, 2005.

BLACK, Alan W.; ZEN, Heiga; TOKUDA, Keiichi. **Statistical parametric speech synthesis**. In: ICASSP 2007. 2007.

BORGES, José Antônio. **Manual do Sistema Operacional Dosvox**. Versão 3.2. Núcleo de Computação Eletrônica - Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2005.

BRANDÃO, Alexandre de Souza. **Modelagem acústica da produção da voz utilizando técnicas de visualização de imagens médicas associadas a métodos numéricos**. Tese de doutorado em Engenharia Mecânica. 172p. Universidade Federal Fluminense. Niterói, 2011.

BRASIL – CONSTITUIÇÃO FEDERAL. Disponível em: < **http://www.planalto.gov.br/ccivil_03/constituicao/constituicaocompilado.htm**>. Acesso: Janeiro 2015.

CHEN, Yan-You; KUAN, Ta-Wen; TSAI, Chun-Yu; WANG, Jhing-Fa; CHANG, Chia-Hao. **Speech variability compensation for expressive speech synthesis**.

COSTA, Ericson Sarmento; MONTE, Anderson de Oliveira; NETO, Nelson; KLAUTAU, Aldebaro. **Um sintetizador de voz baseado em HMMs livre: dando novas vozes para aplicações livres no português do Brasil**. In: Workshop de Software Livre, 2012.

COSTA, Rodrigo Carvalho Souza. **Um novo algoritmo para interação homem-dispositivo portável multiplataforma baseado em fluxo óptico**. Tese de doutorado. Universidade Federal do Ceará. Fortaleza, 2012.

DUTOIT, T.: **An Introduction to Text-To-Speech Synthesis**. Kluwer Academic Publishers, Dordrecht Hardbound. ISBN 0-7923-4498-7, 312 pp. 1997.

DUTOIT, T., H. LEICH, H.: **MBR-PSOLA: Text-To-Speech Synthesis based on an MBE Re-Synthesis of the Segments Database**. In: Speech Communication, Elsevier Publisher, vol. 13, n03-4. 1993.

EICHNER, Matthias; WOLFF, Matthias; OHNEWALD, Sebastien; HOGGMANN,Rüdiger. **Speech synthesis using stochastic markov graphs**. 2001.

ESPEAK - SITE OFICIAL. Disponível em: <**http://espeak.sourceforge.net/**>. Acesso em: Junho de 2014.

FESTIVAL – SITE OFICIAL. Disponível em: <**http://www.cstr.ed.ac.uk/projects/festival/**>. Acesso em: Junho de 2014.

FULKERSON, Michael S.; BIERMANN, Alan W. **Javox: A Toolkit for Building Speech-Enabled Applications**.

GONÇALVES, Maria Inês Rebelo; PONTES, Paulo Augusto de Lima, VIEIRA, Vanessa Pedrosa, PONTES, Antônio de Lima; CURCIO, Daniella; DE BIASE, Noemi Grigoletto. **Função de transferência das vogais orais do Português brasileiro: análise acústica comparativa**. Brazilian Journal of Otorhinolaryngology. Vol. 75, ed. 5 (setembro - outubro). 2009.

HAYKIN, Simon S.; VEEN, Barry Van. **Sinais e Sistemas.** Bookman. 2001.

HUNT, Andrew J.; BLACK, Alan W. **Unit selection in a concatenative speech synthesis system using a large speech database**. 1996.

INSTITUTO BENJAMIN CONSTANT. Disponível em: <**http://www.ibc.gov.br**>. Acesso em: Janeiro de 2015.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. Disponível em: <**http://www.ibge.gov.br/home/estatistica/populacao/censo2010/caracteristicas_religiao_deficiencia/caracteristicas_religiao_deficiencia_tab_xls.shtm**> Acesso em: Dezembro de 2014.

JUNG, Jong-Soon; KIM, Jeong-jin; BAE, Myung-jin. **Pitch alteration technique in speech synthesis system**. In: IEEE Transactions on Consumer Electronics, Vol. 47, No 1. Fevereiro 2001.

KANG, Guangyu; GUO, Shiz; YU, Longjiang. **Speech synthesis algorithm of co-articulation based on the continuous transition of energy**. In: 2009 World Congress on Computer Science and Information Engineering. 2009.

KNOPPER, Klaus. **Desktop Auditivo**. Revista do Linux. 51a Edição. Fevereiro de 2009.

KOBAYASHI, Mei; SAKAMOTO, Masaharu; SAITO, Takasi; HASHIMOTO, Yasuhide; NISHIMURA, Masafumi; SUZUKI, Kazuhiro. **Wavelet analysis used in text-to-speech synthesis**. In: IEEE Transactions on Circuits and Systems - II Analog and Digital Signal Processing, Vol. 45, No. 8. August 1998.

LATHI, B. P. **Sinais e Sistemas Lineares**. 2a Edição. Bookman. 2007.

LIANE TTS - SITE OFICIAL. Disponível em: <**http://intervox.nce.ufrj.br/lianetts/**>. Acesso em: Setembro de 2014.

LIMA, Gislânia Maria de Souza. **Modelagem matemática da voz humana: um exemplo de aplicação de um modelo fonte-filtro**. Monografia em Física. Universidade Estadual do Ceará. Fortaleza, 2010.

LOPEZ, Fernando Carrara; FANGANIELLO, Renato Dalto. **Síntese e predição de sinais de voz**. Trabalho de Graduação Interdisciplinar em Engenharia Elétrica. Universidade Presbiteriana Mackenzie. 55p. São Paulo, 2009.

MACHADO, Cristiano Gaspar. **Um sistema de síntese de voz para a língua portuguesa**. Universidade Federal do Rio de Janeiro. 1997.

MAEDA, Shinji. **Vocal-tract acoustics and speech synthesis**. 1995.

MARANGONI, Josemar Barone; PRECIPITO, Waldemar Barilli. **Reconhecimento e Sintetização de Voz Usando Java Speech**. In: Revista Científica Eletrônica de Sistemas de Informação (ISSN 1807 - 1872). Ano 2, Número 4. 2006.

MATUCK, Gustavo Ravanhani. **Processamento de sinais de voz padrões comportamentais por redes neurais artificiais**. Relatório Final de Projeto de Iniciação Científica. 56p. Instituto Nacional de Pesquisas Espaciais. São José dos Campos 2005.

MBROLA – SITE OFICIAL. Disponível em: <**http://tcts.fpms.ac.be/synthesis/mbrola.html**>. Acesso em: Junho de 2014.

MONTILHA, Rita de Cassia Ietto; TEMPORINNI, Edméa Rita; NOBRE, Maria Inês Rubo; JOSÉ, Newton Kara. **Percepções de escolares com deficiência visual em relação ao seu processo de escolarização**. In: Paideia, vol. 19, No. 44. 2009.

MOORE, Keith L.; DALLEY II, Arthur F. **Anatomia orientada para clínica**. 4a edição. Guanabara Koogan. 2001.

NAÇÕES UNIDAS NO BRASIL. Disponível em: <**http://www.onu.org.br/oms-afirma-que-existem-39-milhoes-de-cegos-no-mundo/**>. Acesso em: Junho 2014.

OPPENHEIM, Alan v.; WILLSKY, Alan S.; NAWAB, S. Hamid. **Signals and Systems**. 2a Edição. Prentice Hall. 1997.

OPPENHEIM, Alan V.; SCHAFER, Ronald W. **Digital Signal Processing**. Prentice Hall International. 1975.

OPPENHEIM, Alan V., SCHAFER, Ronald W. **Discrete-Time Signal Processing**. Prentice-Hall, 2009.

O'SHAUGHNESSY, Douglas. **Interacting with compters by voice: automatic speech recognition and synthesis**. In: Proceedings of the IEEE, Vol. 91, No. 9. Setembro de 2003.

PHUNG, Trung-Nghia; LUONG, Mai Chi; AKAGI, Masato. **A concatenative speech synthesis for monosyllabic languages with limited data**.

PITT, Ian J., and ALISTAIR DN Edwards. **Improving the usability of speech-based interfaces for blind users**. In: Proceedings of the second annual ACM conference on Assistive technologies. ACM, 1996.

PUTZ, R. e PABST, R. **Sobotta - Atlas de Anatomia Humana. Volume 1 - Cabeça, pescoço e extremidade superior**. 21a Edição. Guanabara Koogan. 2001.

SÁNCHEZ, Jaime; AGUAYO, Fernando. **APL: Audio Programming Language for Blind Users**. In: VII Congresso Iberoamericano de Informática Educativa. 2004.

SANTOS, Andréa dos; FRANÇA, Halisson Fabrício de Carvalho; GOMES, Ítalo Herbert Santos e; TEIXEIRA, Wander Glayson Fernandes; FILHO, Guido Lemos de Souza. **Desenvolvimento de aplicações para Deficientes Visuais: Uma discussão sobre Ferramentas para Incorporação da Tecnologia de Voz ao VoiceProxy**. Universidade Federal do Rio Grande do Norte.

SANTOS, Jader Gustavo de Campos. **Acessibilidade em aplicações desktop utilizando ferramentas Java**. Monografia de Especialização. Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2010.

SCHUMACHER; ROBERT M.; HARDZINSKI, Mary L.; e SCHWARZ, Amy L. **Increasing the usability of interactive voice response systems: Research and guidelines for phone-based interfaces**. Human Factors: The Journal of the Human Factors and Ergonomics Society 37.2 251-264. 1995.

SCHROETER, Juergen. **Electrical Engineering Handbook.** Capítulo 16: Circuits, Signals, Speech and Image Processing. 3ª. Edição. AT&T Laboratories. 2005.

SECRATARIA DE DIREITOS HUMANOS DA PRESIDÊNCIA DA REPÚBLICA. **Cartilha do Censo 2010 - Pessoas com Deficiência**. 32p. Brasília, 2012.

SHU, Chang; MEI, Jin-Shuo, YIN, Jing-Hua. **Speech synthesis based on AMR-WB algorithm**. In: 2011 International Conference on Electronic & Mechanical Engineering and Information Technology. Agosto 2011.

SPANIAS, Andreas S. **Speech Coding: A Tutorial Review**. In: Proceedings of the IEEE, Vol. 82, No 10. 1994.

SUN MICROSYSTEMS. **GNOME 2.0 Desktop: Developing With the Accessibility Framework**. Sun Microsystems. 2003.

SUN MICROSYSTEMS. **Java™ Speech API Programmer's Guide**. Versão 1.0. Sun Microsystems. Palo Alto, Outubro de 1998.

SUN MICROSYSTEMS. **Java™ Speech Grammar Format Specification**. Versão 1.0. Sun Microsystems. Palo Alto, Outubro de 1998.

SUN MICROSYSTEMS. **Java Speech Markup Language Specification**. Versão 0.5. Sun Microsystems. Mountain View, Agosto de 1997.

TABET, Youcef; BOUGHAZI, Mohamed. **Speech synthesis techniques: A survey**. In: 7th International Workshop on Systems, Signal Processing and Their Applications (WOSSPA). 2011.

TALAFOVÁ, R.; ROZINAJ G.; CEPKO, J. **Speech synthesis for mobile phone**. In: 49 International Symposium ELMAR-2007. Zadar, Croatia, 2007.

TAMURA, Masatsune; BRAUNSCHWEILER, Norbert; KAGOSHIMA, Takehiko; AKAMINE, Masami. **Unit selection speech synthesis using multiple speech units at non-adjacente segments for prosody and waveform generation**. In: ICASSP 2010. 2010.

WALKER, Mark R.; LARSON, Jim; HUNT, Andrew. **A new W3C markup standard for text-to-speech synthesis**. 2001.

WOUTERS, Johan; MACON, Michael W. **Spectral modification for concatenative speech synthesis**. 2000.

YANKELOVICH, Nicole; LEVOW, Gina-Anne; e, MARX, Matt. **Designing SpeechActs: Issues in speech user interfaces.** Proceedings of the SIGCHI conference on Human factors in computing systems. ACM Press/Addison-Wesley Publishing Co. 1995.

APÊNDICE A: MODELAGEM MATEMÁTICA DO TRATO VOCAL

A.1 O trato vocal

O trato vocal, mostrado nas Figura A.1 e A.3, é composto pela laringe e faringe, ou cavidades laríngea e faríngea, respectivamente, cavidades oral, também chamada de cavidade bucal e mostrada na Figura A.2, e cavidade nasal, tendo início, portanto na abertura entre as pregas vocais, uma fibra elástica com duas pregas que se distende ou relaxa pela ação de músculos no interior da laringe, glote, e terminando nos lábios.

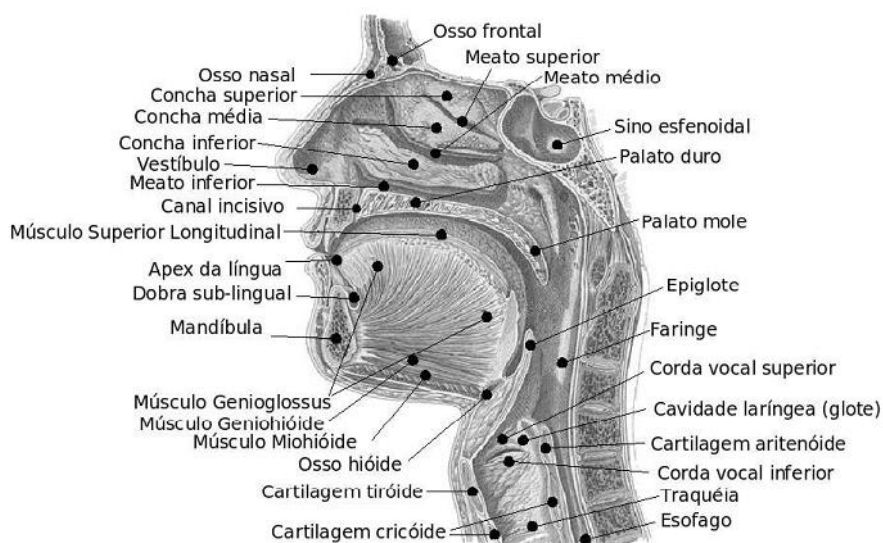



Figura A.1: Trato vocal em detalhes. Fonte: (BRANDÃO, 2011).

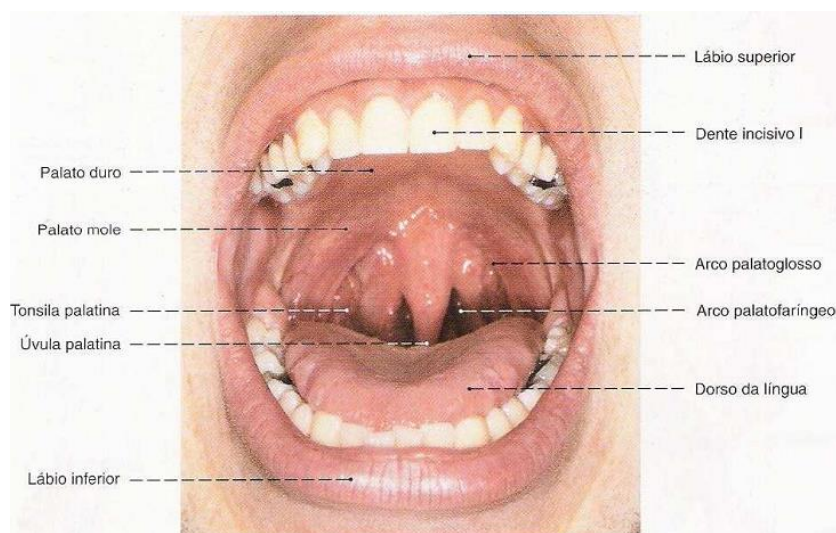



Figura A.2: cavidade própria da boca. Vista ventral. Fonte: (PUTZ, 2001).

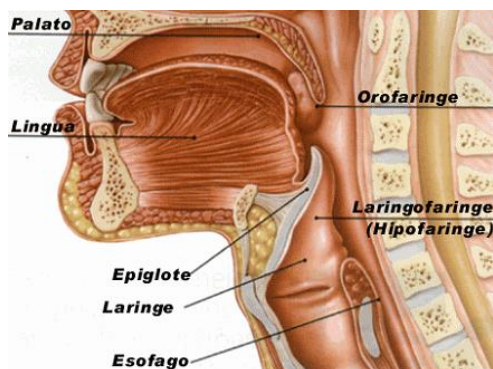



Figura A.3: anatomia da garganta. Fonte: (MATUCK, 2005).

O comprimento médio do trato vocal masculino é de aproximadamente 17 cm, sendo que este valor praticamente não varia, com área de seção transversal determinada pela posição da língua, lábios, maxilar e véu palatino, variando entre zero, o fechamento completo, até 20 cm^2, assumindo, portanto, diferentes formas. A modificação da forma do trato vocal permite a diversificação do som e é realizado pela língua (BRANDÃO, 2011; LIMA, 2010).

O trato nasal inicia-se no véu palatino e termina nas narinas. Quando o véu palatino baixa, o trato nasal é acoplado acusticamente ao trato vocal, cujas cavidades constituem a estrutura ressoadora do órgão da voz, tendo função semelhante à dos ressonadores de instrumentos musicais (BRANDÃO, 2011; LIMA, 2010).

A teoria aerodinâmica-mioelástica postula que o movimento de abrir e fechar as pregas vocais são regidos por propriedades mecânicas dos tecidos musculares que constituem, principalmente, as pregas vocais e pelas forças aerodinâmicas que se distribuem ao longo da laringe durante a fonação. A ação neural consiste apenas em aproximar as pregas vocais de tal forma que a superfície destas vibre (LIMA, 2010).

O conjunto de músculos responsáveis pela movimentação das pregas vocais é mostrado na Tabela A.1. Os movimentos de elevação e de depressão da laringe são controlados respectivamente pelos músculos extrínsecos elevadores e depressores. Por sua vez os músculos intrínsecos controlam a posição e a tensão das pregas vocais, para depois, o sinal deve ser amplificado pelo trato vocal que pode ser analisado a partir de um modelo de tubos simples (LIMA, 2010).

Tabela A.1: músculos responsáveis pela movimentação das pregas vocais e órgãos relacionados.

Músculo	Ação Principal
Cricotireoideo	Estica e tensiona a prega vocal.
Cricoaritenóideo posterior	Abduz a prega vocal.
Cricoaritenóideo lateral	Abduz a prega vocal (porção interligamentosa).
Tireoaritenóideo	Relaxa a prega vocal.
Aritenóideos oblíquo e transverso	Fecha a porção intercartilagínea da rima da glote.
Vocal	Relaxa a parte posterior do ligamento vocal enquanto mantendo (ou aumentando a tensão da parte anterior).

Fonte: (MOORE e DALLEY, 2001).

O trato vocal funciona como um guia de onda ou filtro acústico que deixa passar o sinal sonoro produzido pela vibração das pregas vocais em determinadas frequências, enquanto atenua outras. Tal vibração das pregas vocais resulta na produção de voz e é resultante do fluxo de ar proveniente dos pulmões, ocasionando o chamado Efeito Bernoulli, mostrado na Figura A.4 (BRANDÃO, 2011).

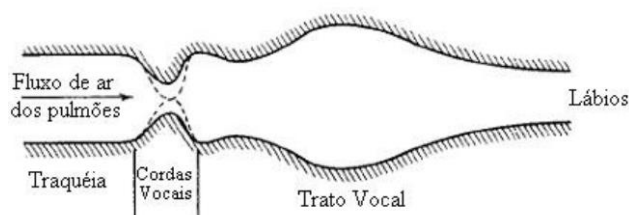



Figura A.4: efeito de Bernoulli nas pregas vocais.

No processo de emissão de voz, há cinco fenômenos relacionados: a respiração, a fonação, a ressonância, a articulação e a prosódia. Na respiração, o fôlego e o controle respiratório são importantes para que não ocorra interrupção durante a fala. O fenômeno da fonação se refere à qualidade e às características da voz produzida pela laringe enquanto que a ressonância é a modificação seletiva da inflexão na voz quando a corrente de ar passa através da rinofaringe, orofaringe e boca. Durante este processo, ocorre a modulação ou amplificação da voz, que cria características individuais da voz. A articulação é outro fenômeno, sendo resultante do movimento dos lábios, língua, dentes, palato duro ou mole. Trata-se da produção de sons da fala por meio da parada ou constrição do fluxo de ar, vocalizado ou não, por meio de tais movimentos destes referidos órgãos. Por fim, a prosódia se refere à velocidade, intervalo, melodia e ênfase (MATUCK 2005).

Os órgãos responsáveis pela fonação são: laringe, lábios, língua, dentes, véu palatino e boca, mostrados na Figura A.5.

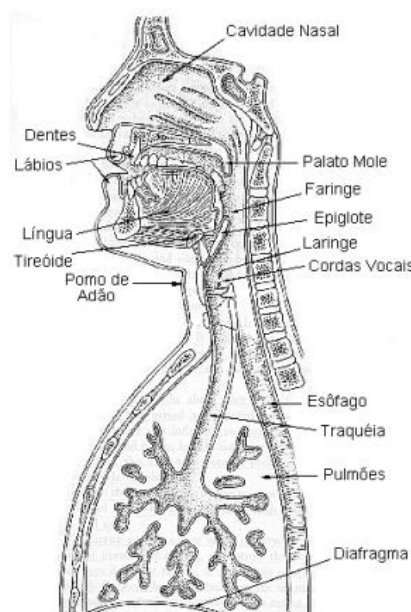



Figura 2.5: órgãos responsáveis pela fonação. Fonte: (MACHADO, 1997).

É possível classificar os órgãos atuantes na fonação em cinco grupos: o da respiração, o da vocalização, o da ressonância, o da articulação e o de irradiação, estes são ilustrados na Figura A.6. O grupo da respiração é responsável pela produção de um fluxo de ar, iniciando nos pulmões e terminando na traqueia; já o de vocalização é responsável produção do sinal glotal, ocorrendo na faringe. Este sinal é de baixa intensidade que necessita ser amplificado para que determinadas componentes harmônicas sofram "ênfase", de maneira que os fonemas sejam caracterizados. Tal fenômeno é realizado pelo grupo de ressonância, composto pela faringe, cavidades oral e nasal, e ocorre na passagem do ar impulsionado nos pulmões pelo trato vocal. Além disso, filtra os pulsos de ar gerados pela vibração das pregas vocais. Já o sistema articulador modifica as propriedades de filtragem dos órgãos de ressonância sobre o sinal glotal, irradiando o som para o meio externo, cuja frequência dos pulsos de ar que passam pelo trato vocal determina basicamente o quão agudo ou grave é uma voz. Ao chegar à boca, tais as ondas de pressão são irradiadas, sendo esta tarefa realizada pelo grupo de irradiação (LIMA, 2010; MACHADO, 1997).

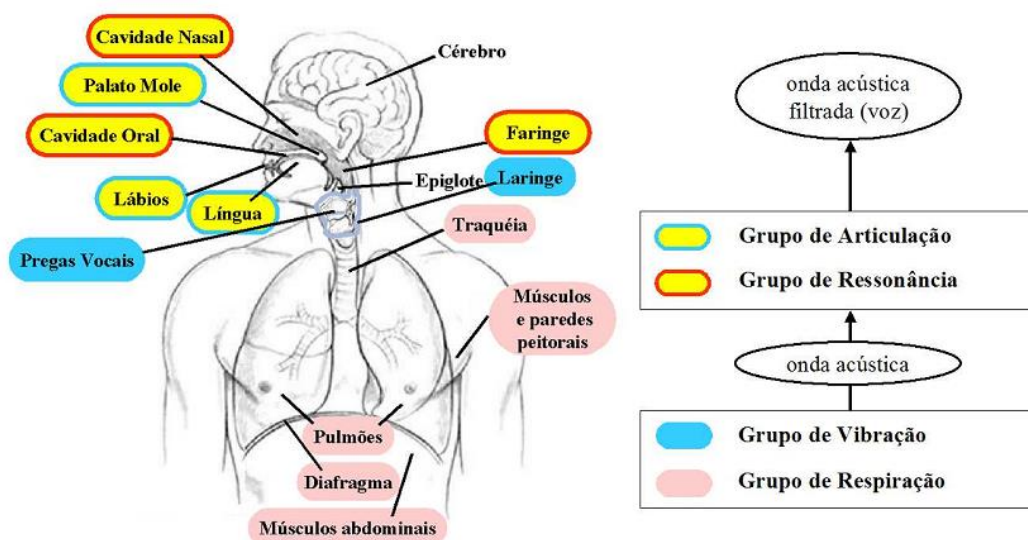



Figura A.6: esquema de produção da voz humana. Fonte: (BRANDÃO, 2011).

A produção da voz se inicia, portanto, com uma contração-expansão dos pulmões, criando assim, uma diferença de pressão entre o ar nos pulmões e o ar na frente da boca, causando um deslocamento de ar. Tal deslocamento passa pela laringe, transformando-se em uma série de pulsos, o sinal glotal, que chegam à boca e à cavidade nasal, sendo modulados pelas línguas, dentes e lábios (LIMA, 2010).

Com o aumento da pressão nos pulmões, o ar flui para fora destes e atravessa as pregas vocais (glote). De acordo com a lei de Bernoulli, quando um fluido se desloca por um orifício, a pressão é menor na constrição do que nas áreas adjacentes. Se a tensão nas pregas vocais for adequada, a pressão reduzida permite que as pregas vocais se toquem, bloqueando completamente o fluxo de ar. Como resultado deste bloqueio no fluxo de ar, a pressão sob as pregas vocais aumenta até finalmente atingir um nível suficiente para forçar abertura das pregas vocais e, assim, permitir o fluxo de ar através da glote. A pressão na glote cai novamente e o ciclo se repete (LOPEZ, 2009).

Desta forma, durante a fonação, as pregas vocais entram em uma condição de oscilação sustentada. A taxa com que a glote abre e fecha é controlada pela pressão de ar nos pulmões, pela tensão nas pregas vocais e pela rigidez das mesmas, além da área de abertura da glote na condição de repouso. Estes são os parâmetros de controle de um modelo para o comportamento das pregas vocais. Tais modelos devem também conter a influência do trato vocal, uma vez que as variações de pressão no trato vocal interferem nas variações de pressão na glote (LOPEZ, 2009).

Tais ciclos vibratórios se repetem muitas vezes por segundo, dependendo da pessoa e da tensão aplicada nas pregas vocais, formando o sinal glotal. Tal frequência corresponde à frequência fundamental, também chamada de *pitch*, e representa o período de interrupção do fluxo de ar que excita o trato vocal causado pela vibração das pregas vocais quando passado pela glote.

Considerando que a cada ciclo a glote abre devagar e fecha muito rápido, isso deve fazer com que o trem de pulsos de onda de pressão tenha um ataque lento e uma queda rápida. No domínio do tempo, a pressão $P(t)$ é definida por (LIMA, 2010)

$$P(t) = P_0 \left(\frac{t}{T}\right)^{\alpha} sen^2 \left(\pi \frac{t}{T}\right), \quad (1)$$

em que T é o período, α nos diz se o ataque é lento ou rápido. Quanto maior alfa, mais inclinado é o pulso, caso α seja nulo, não há inclinação.

Conforme dito anteriormente, a frequência de vibração das pregas vocais durante a fonação pode ser modificada pelos músculos laríngeos e pressão do ar gerada pelos pulmões. Em resposta à variação de tensão dos músculos, as pregas vocais vibram a frequências de 50 a 1000Hz, resultando em sopros de ar injetado na traqueia. Quanto maior for esse período, menor é o espaço entre as harmônicas e, consequentemente menor é o seu período fundamental, resultando em um som mais grave. Por outro lado, se esse período for muito pequeno, a frequência fundamental é alta, logo, produzindo som mais agudo (BRANDÃO, 2011; LOPEZ, 2009).

A alteração da frequência fundamental é realizada de tal forma que as informações linguísticas são fornecidas ao interlocutor através da entonação, indicando perguntas, afirmações ou estados emocionais. As componentes de frequência do sinal de voz que são enfatizadas para uma determinada configuração do trato vocal são denominadas de formantes, um conjunto composto por quatro ou cinco ressonâncias importantes que formam uma zona de alta concentração de energia acústica. Diferentes combinações de frequências formantes são geradas conforme formato assumido pelo trato vocal, gerando diferentes sons vozeados (BRANDÃO, 2011, LIMA, 2010).

A frequência natural da voz é influenciada também pelo comprimento das pregas vocais: mulheres e crianças apresentam vozes mais agudas porque suas pregas vocais são mais curtas (MATUCK, 2005), cuja movimentação durante a fonação é mostrada nas Figuras A.7 até A.11. Na Figura A.10 é possível observar o abaulamento da prega vestibular.

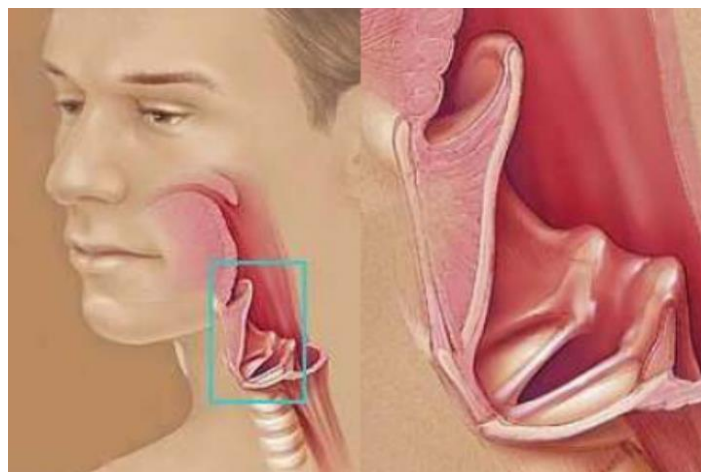



Figura A.7: localização das pregas vocais. Fonte: (LOPEZ e FANGANIELLO, 2007).

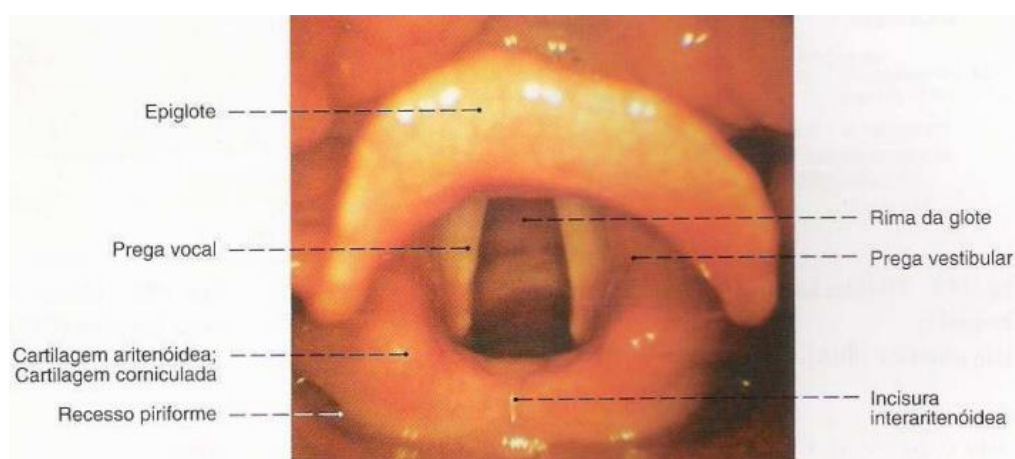



Figura A.8: laringoscopia direta - pregas vocais na respiração profunda. Posição respiratória. Fonte: (PUTZ, 2001).

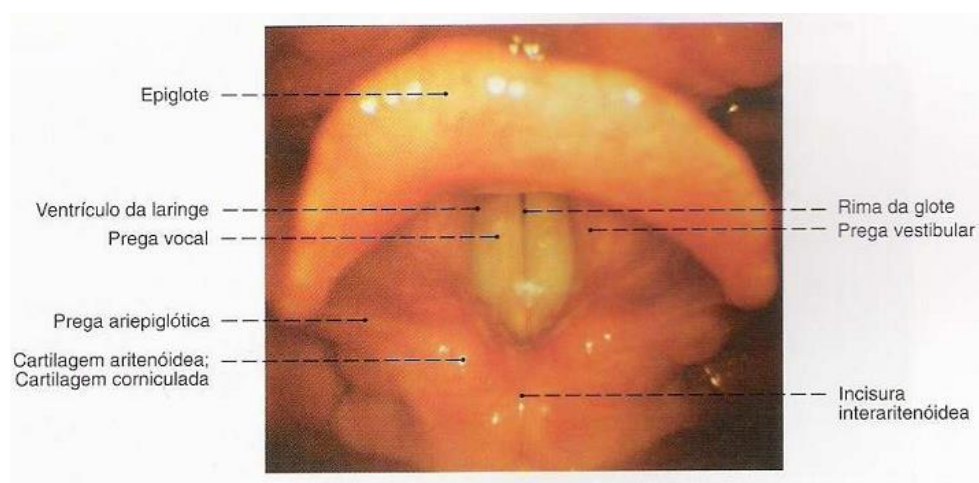



Figura A.9: laringoscopia direta - pregas vocais fechadas. Posição de fonação. Fonte: (PUTZ, 2001).

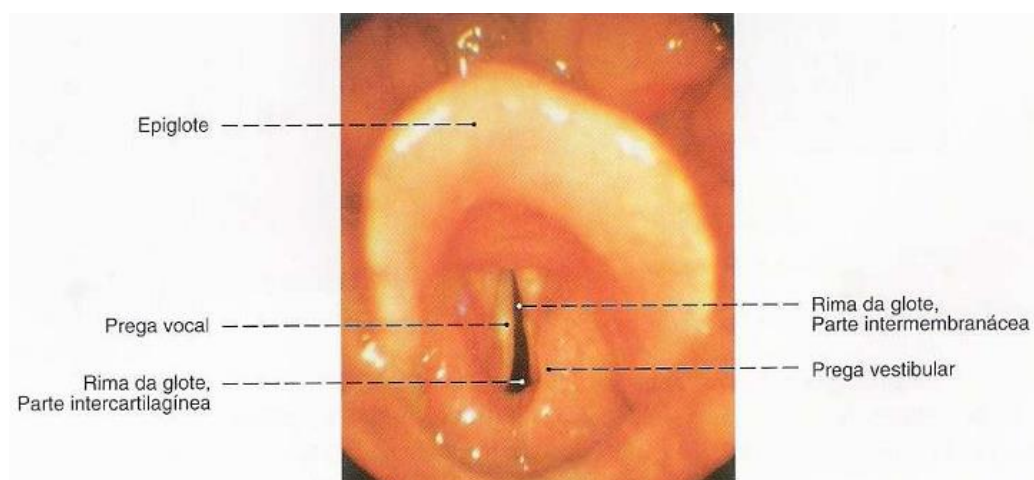



Figura A.10: laringoscopia direta - Parte intercartilagínea da glote aberta na posição de cochicho. Fonte: (PUTZ, 2001).

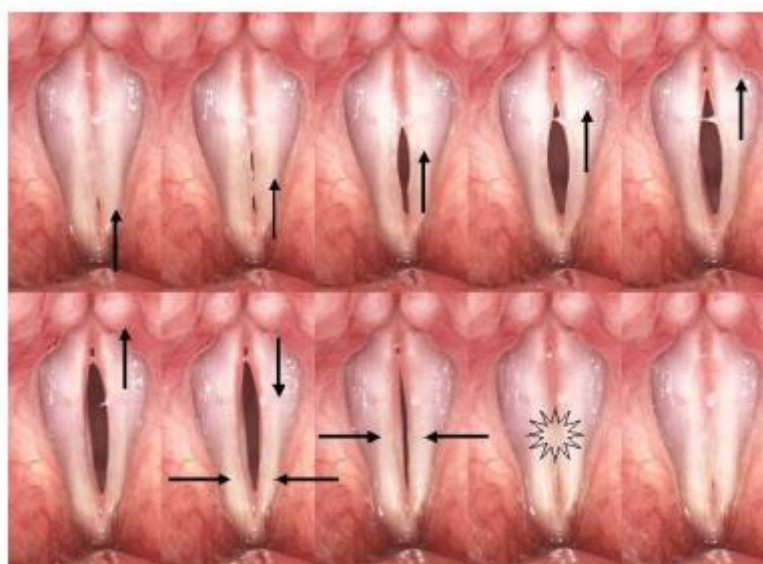
(a)

(b)

Figura A.11: (a) Movimentação das pregas vocais durante a fonação. (b) Imagem real de uma prega vocal durante a fonação. Fonte (BRANDÃO, 2011).

Cada prega ou corda vocal é uma dobra de tecidos que se comportam como um conjunto mecânico composto por músculo e ligamentos rígidos e pesados revestidos por um conjunto composto por tecido conjuntivo e epitelial flexível. O fluxo de ar é modulado à medida em que as pregas vocais abrem e fecham ciclicamente. A vibração gotal ocorre de forma aproximadamente periódica, mas com velocidade de fechamento maior que a de abertura em cada ciclo, que permite o aparecimento de uma componente harmônica além da fundamental (LIMA, 2010). A dinâmica das pregas vocais é mostrada na Figura A.12, na qual está mostrado um ciclo completo (LIMA, 2010).

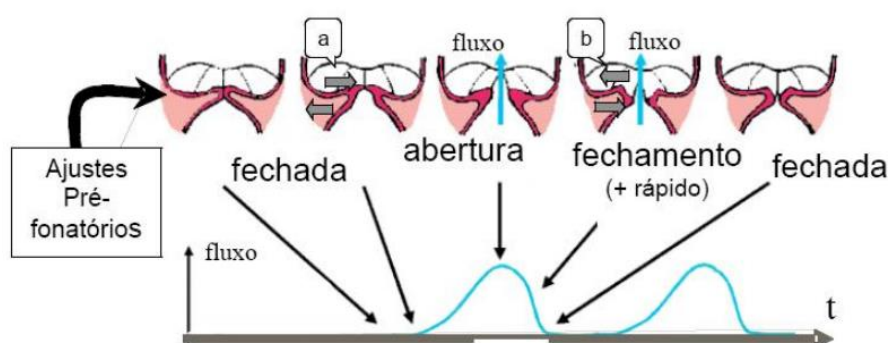



Figura A.12: ciclo fonatório. Fonte: (LIMA, 2010).

Dependendo da frequência e intensidade do som que se deseja produzir, podem ser realizados ou não ajustes de pressão pré-fonatórios abaixo da glote e provocada pelo ar dos pulmões, denominada pressão sublótica, pela tensão longitudinal, pela aproximação da parte posterior das pregas vocais e da força de compressão na parte medial. Com o esforço expiatório e com a glote fechada ainda, a pressão intraglótica aumenta enquanto que as bordas inferiores se afastam e acumulam energia potencial elástica na camada de abertura. Tal aumento da pressão intraglótica faz com que as bordas superiores se separem, permitindo que o ar flua pela glote. Tal fluxo leva a uma queda da pressão, que ocorre em um momento em que as bordas inferiores estão comprimidas, resultando em um fechamento mais rápido que a abertura, ocasionando a assimetria em um ciclo que se repete na frequência fundamental (LIMA, 2010).

O processo de fala é um processo retroalimentado, mostrado na Figura A.13, no qual, para que haja uma correta fala, é necessário realimentar o aparelho fonador com o som produzido a fim de que possa realizar eventuais ajustes biomecânicos necessários.

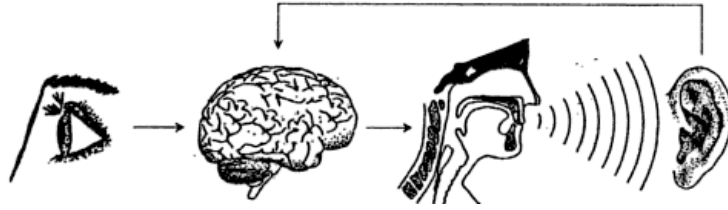



Figura A.13: fluxo do processo de leitura e fala como um processo retroalimentado. Fonte: (DUTOIT, 1997).

A.2 Modelagem matemática das ondas sonoras

A propagação de uma onda acústica pode ser aproximada considerando a propagação de perturbações infinitesimais em um fluido compressível sem viscosidade. A equação de onda descreve o movimento da onda em um meio através da evolução da pressão acústica p, ou da velocidade u, da partícula em função da posição $\boldsymbol{r} = (x, y, z)$ e do tempo t sendo dada pelas Equaçoes 16 e 17:

$$\frac{\partial^2 p(\boldsymbol{r}, t)}{\partial t^2} - c^2 \nabla^2 p(\boldsymbol{r}, t) = 0, \tag{16}$$

$$\frac{\partial^2 u(\boldsymbol{r}, t)}{\partial t^2} - c^2 \nabla^2 u(\boldsymbol{r}, t) = 0, \tag{17}$$

em que $p(\boldsymbol{r}, t)$, $u(\boldsymbol{r}, t)$, ρ e c são perturbações na pressão estática e na velocidade da partícula de ar, a densidade do ar e a velocidade do som no ar, respectivamente, em um ponto $\boldsymbol{r} = (x, y, z)$ do espaço tridimensional no instante de tempo t (BRANDÃO, 2011).

A velocidade c da propagação acústica no ar é dada pela Equação 18:

$$c = \sqrt{\frac{\gamma p_0}{\rho}}, \tag{18}$$

em que $\gamma = \frac{c_p}{c_v}$ é a razão entre os calores específicos do ar a pressão e volume constantes, p_0 e ρ são a pressão atmosférica e a densidade do ar, respectivamente (BRANDÃO, 2011).

Para este estudo, podemos desconsiderar eventuais variações da temperatura do ar no interior do trato vocal por serem muito pequenas, bem como densidades. Estudos mostram que a turbulência do fluxo de ar que passa pelo trato vocal durante a vocalização pode ser desconsiderada (BRANDÃO, 2011).

A equação da onda na forma clássica é dada pela Equação 19:

$$\frac{\partial^2 P'}{\partial t^2} = c^2 \nabla^2 P', \tag{19}$$

em que c é a velocidade de propagação de um som no fluido e P' é uma pequena perturbação no fluido, cuja solução pode ser obtida por meio de separação de variáveis chegando-se à equação escalar de Helmholtz e é dada pela Equação 20:

$$\Phi(t) = Acos\omega t + Bsen\omega t, \tag{20}$$

(LIMA, 2010).

A Equação de Movimento de Euler para o movimento de um fluido ideal sob a existência de uma força externa F em um meio fluido de densidade ρ é dada pela Equação 21:

$$\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v}\nabla)\boldsymbol{v} + \frac{1}{\rho}\nabla P = \frac{\boldsymbol{F}}{\rho}, \tag{21}$$

(LIMA, 2010).

A.3 Modelagem matemática do trato vocal

Para intervalos curtos de tempo, de 3 a 40ms, pode-se modelar a voz por meio de três parâmetros: (a) a seleção de excitação por sequência de impulsos periódica ou por ruído gaussiano, (b) a frequência fundamental (*pitch*) da excitação periódica, quando utilizada e (c) os coeficientes de um filtro recursivo linear simulando o trato vocal, cujo esquema é representado pelo diagrama de blocos ilustrado na Figura A.14. Pode-se então, sintetizar voz atualizando-se continuamente estes parâmetros cerca de 40 vezes por segundo. Embora a qualidade sonora desta aproximação seja baixa, soando mecânico em vez de humano, requer baixa taxa de atualização de dados (LOPEZ e FANGANIELLO, 2007).

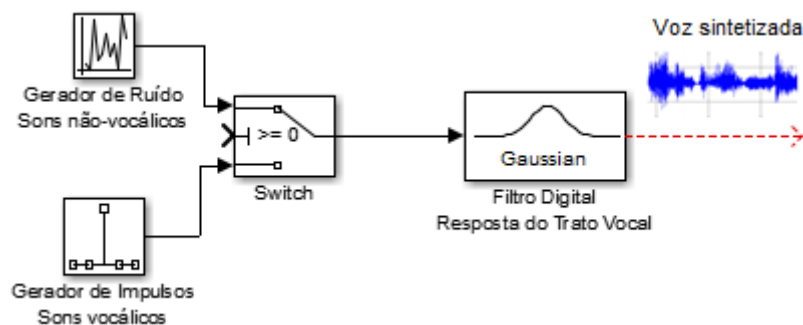



Figura A.14: diagrama de blocos de um sintetizador de voz genérico.

Um modelo detalhado do sistema vocal deve envolver pulmões, brônquios, traqueia, glote e o trato vocal. O primeiro trabalho abrangente em busca de um modelo físico detalhado para a geração de som no trato vocal foi realizado no final da década de 1960.

Pesquisas subsequentes produziam um modelo mais refinado, fornecendo representação mais detalhada do processo de geração de sons vozeados e não-vozeados. Tal modelo se baseia em mecânica clássica e mecânica dos fluidos (LOPEZ e FANGANIELLO, 2007).

Utilizam-se duas aproximações para geração de voz: gravação digital e simulação do trato vocal. No caso de gravação digital, a voz de um falante humano é digitalizada e armazenada, geralmente sob uma forma comprimida. Durante a reprodução, os dados armazenados são descomprimidos e convertidos em sinal analógico. Já a simulação do trato vocal é mais complexa, pois tenta imitar o mecanismo físico pelo qual a voz humana é gerada. Podemos tomar inicialmente um sinal $\tilde{s}(n)$ que se trata de um determinado sinal $s(n)$ amostrado. Seja $x(n)$ a entrada e G o ganho, podemos adotar o seguinte modelo mostrado na Equaçao 22:

$$\tilde{s}(n) - \sum_{i=1}^{p} \alpha_i \, \tilde{s}(n-1) = Gx(n), \tag{22}$$

Tipicamente, os coeficientes α_i variam a cada 10 a 20ms de acordo com mudanças do trato vocal para a produção dos diferentes sons. Para a síntese, aplica-se uma sequência de excitação ao modelo que contém os coeficientes apropriados para cada intervalo de tempo a fim de se gerar a sequência de sons desejada. Para o caso, temos o seguinte polinômio característico mostrado na Equação 23:

$$Q(z) = 1 - \sum_{i=1}^{p} \alpha_i \, z^{-1} = \prod_{i=1}^{p-1} (z - z_i), \tag{23}$$

como a equação é de ordem p, temos então p raízes características z_i. Geralmente, para voz masculina, temos $p = 10$ e as raízes formam pares complexos conjugados de forma que todos os coeficientes α_i assumem valores reais (LOPEZ e FANGANIELLO, 2007).

A síntese de voz utilizando o modelo de equações de diferenças requer que, primeiramente, um segmento de voz real seja analisado para que se possa determinar quais coeficientes α_i são mais apropriados para cada segmento de 10ms. Para cada um destes segmentos, deve-se calcular um conjunto de coeficientes α_i. O processo de extração de um bloco de 10ms do sinal original é chamado janelamento. A função de janelamento mais comum é a Janela de Hamming, que apresenta transição mais suave, evitando problemas de análise (LOPEZ e FANGANIELLO, 2007).

Após o janelamento, uma análise estatística dos dados que determina o grau de correlação entre as amostras adjacentes é utilizada para se calcular os coeficientes que

forneça a melhor predição do sinal, isto é, que minimize o erro de predição. Uma vez encontrados estes coeficientes, pode-se sintetizar voz aplicando-se um sinal apropriado de entrada ao modelo. No caso de sons vozeados, um bom modelo da fonte para o sinal de entrada é um trem de impulsos ideais a uma dada frequência, sendo que a frequência determina o *pitch*. Já no caso de sons não vozeados, um bom modelo de fonte para o sinal de entrada é um ruído branco gaussiano (LOPEZ e FANGANIELLO, 2007).

A Figura A.15 mostra a variação espectral do *pitch* para a vogal A.

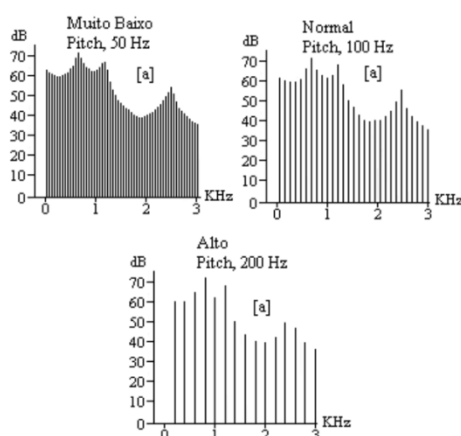



Figura A.15: variação espectral do *pitch* da vogal A. Fonte: Google Images.

A.3.1 Linhas de Transmissão

A geração e propagação dentro do trato vocal pode ser simulada por meio de linhas de transmissão acústicas, mostradas na Figura A.16. Os valores dos elementos acústicos dessa linha de transmissão podem ser descritos por meio da variação das seções transversais ao longo do trato vocal. A função transferência (relação saída entrada) desse sistema, no caso, se dá pelas relações entre som irradiado e fonte. Entretanto, o custo computacional dessa metodologia é muito maior do que os dos sintetizadores baseados em síntese de formantes (MAEDA, 1995).

No domínio acústico, vogais orais são caracterizadas apenas pelos polos na função de transferência, enquanto que consoantes requerem polos e zeros. A interpolação no domínio acústico então se torna complicado caso se deseje interpolar a transição entre uma consoante para vogal adequadamente (MAEDA, 1995).

Existe uma analogia entre as ondas de pressão e as ondas elétricas, tal que a pressão equivale à diferença de potencial, ou tensão elétrica, e o escoamento de ar, causado pela diferença de pressão entre dois pontos, equivale à corrente elétrica, que surge quando há diferença de potencial elétrico entre dois pontos. Assim, nos primeiros trabalhos, era

definido um sistema de equações de malha de circuitos elétricos para representar o conjunto de sessões cilíndricas através do qual o trato vocal foi modelado. As seções cilíndricas são representadas por linhas de transmissão (BRANDÃO, 2011).

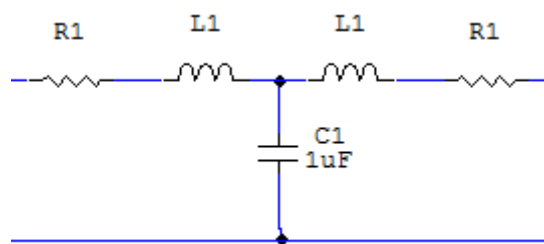



Figura A.16: modelo de uma linha de transmissão. Fonte: (BRANDÃO, 2011 - Adaptado).

A irradiação do som pela boca é modelada como uma impedância de radiação de forma similar à modelagem de uma antena em sistemas elétricos, formada por uma resistência R_τ e por uma indutância L_t em paralelo. O som sintetizado corresponde à diferença de pressão entre os terminais dessa impedância. Modelando matematicamente as funções dos quatro grupos do sistema fonador humano, chegou-se ao circuito equivalente acústico mostrado nas Figuras A.17 e A.18 (BRANDÃO, 2011).

A solução numérica das equações correspondentes ao circuito, para cada instante da amostragem, gera uma sequência de valores que representa a voz sintetizada. Mojhatari construiu um modelo de trato vocal em linhas de transmissão quer permite a inclusão de um número indefinido de ramificações para representar reentrâncias do trato vocal e do trato nasal (BRANDÃO, 2011).

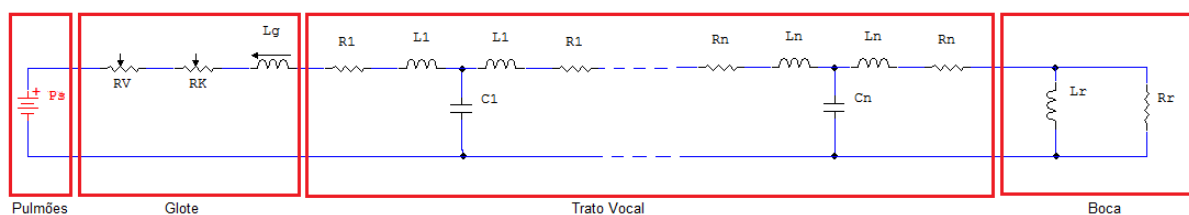



Figura A.17: modelo de linha de transmissão aplicado ao trato vocal. Fonte: (BRANDÃO, 2011 - Adaptado).

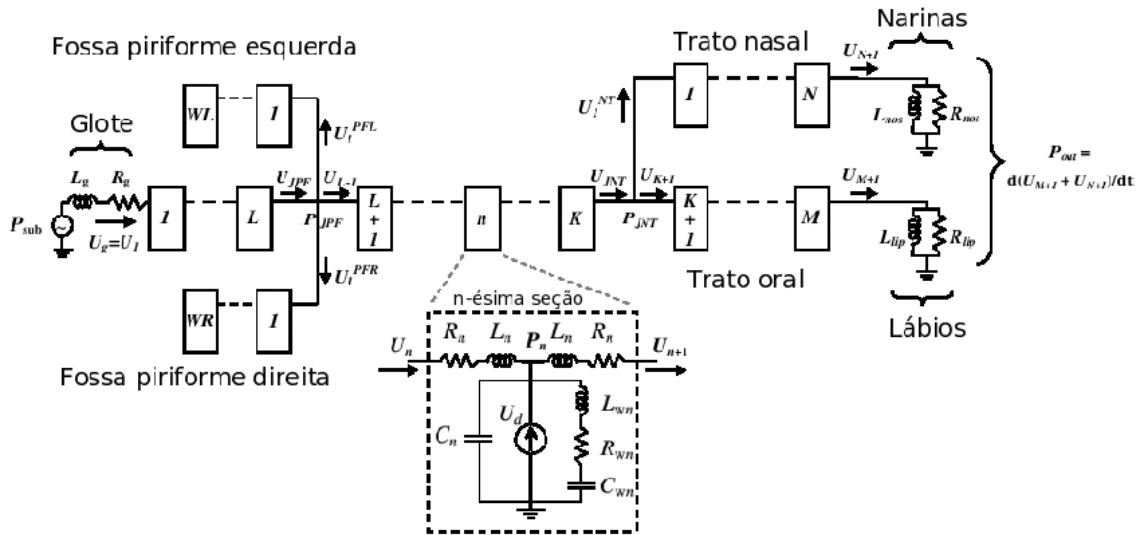



Figura A.18: diagramas esquemáticos, de blocos e de fluxo de sinal integrados para a modelagem do trato vocal. Fonte: (BRANDÃO, 2011).

A.3.2 Modelo de Tubos: Caso Contínuo

O modelo Kelly-Lochbaum é um modelo unidimensional que aproxima o trato vocal como sendo uma sequência de tubos, conforme mostrado na Figura A.19, representados por guias de onda digitais (BRANDÃO, 2011).

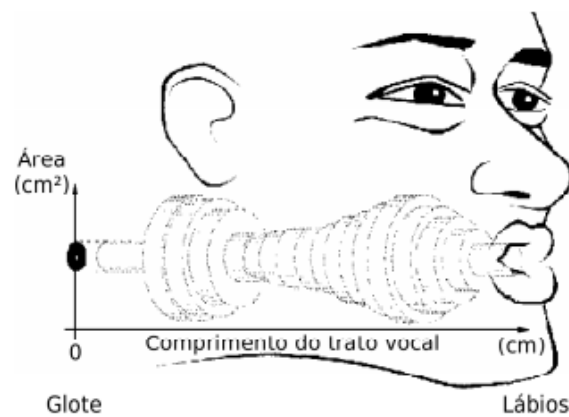



Figura A.19: modelagem do trato vocal. Fonte: (BRANDÃO, 2011).

Como dito anteriormente, sons produzidos pelo ser humano são resultantes da vibração das pregas vocais para fontes vocais ou pelo fluxo de ar turbulento por meio de constrição para fontes de ruído. Tais fontes sofrem modificação espectral por meio das características ressonantes do trato vocal. Uma vez que o trato vocal humano pode ser considerado um tubo, a maior ressonância ocorre ao longo do comprimento da glote até os lábios (ou cavidade nasal no caso de sons nasais) (MAEDA, 1995).

Pode-se modelar o trato vocal e nasal como tubos de secção transversal não uniforme, conforme mostrado na Figura A.20. À medida que o som se propaga através destes tubos, o espectro de frequência é moldado de acordo com a seletividade de frequência do tubo, produzindo um efeito semelhante à ressonância observada em instrumentos de sopro. A frequência de ressonância do trato vocal é chamada de frequência formante ou simplesmente formante.

As frequências formantes dependem do formato e das dimensões do trato vocal, pois formatos diferentes implicam diferentes conjuntos de frequências formantes, podendo-se produzir diferentes sons por meio da alteração do formato do trato vocal. Assim, as propriedades espectrais dos sinais de voz variam com o tempo conforme o formato do trato vocal se altera.

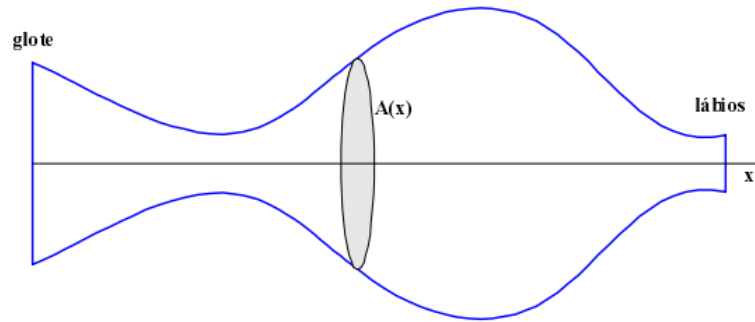



Figura A.20: modelo geométrico genérico do trato vocal. Fonte: Google Images.

Para o referente modelo, deve-se inicialmente pressupor que as seguintes aproximações são válidas: (1) o trato vocal é um tubo acústico linear; (2) a onda sonora é uma onda plana; (3) o meio de propagação é uniforme (ρ constante); (4) as paredes são sem perdas. A partir destas aproximações, é possível demonstrar que as ondas acústicas dentro de um tubo satisfazem as seguintes equações diferenciais parciais mostradas nas Equações 24 e 25:

$$-\frac{\partial p(x,t)}{\partial x} = \rho \frac{\partial \left(\frac{u(x,t)}{A(x,t)} \right)}{\partial t}, \tag{24}$$

$$-\frac{\partial p(x,t)}{\partial x} = \frac{1}{\rho c^2} \frac{\partial \big(u(x,t) A(x,t) \big)}{\partial t} + \frac{\partial A(x,t)}{\partial t}, \tag{25}$$

em que $p(x,t)$ é a pressão acústica no ponto x e no instante t, $u(x,t)$ é o fluxo no ponto x e no instante t, $A(x,t)$ é a área da secção do tubo acústico no ponto x e no

instante t, e ρ é a densidade do ar no interior do tubo e c a velocidade de propagação do som no ar. O sistema acima tem como solução:

$$u(x,t) = u^{+}\left(t - \frac{x}{c}\right) - u^{-}\left(t + \frac{x}{c}\right), \quad (26)$$

$$p(x,t) = \frac{pc}{A}\left(u^{+}\left(t - \frac{x}{c}\right) + u^{-}\left(t + \frac{x}{c}\right)\right), \quad (27)$$

em que $u^{+}\left(t - \frac{x}{c}\right)$ e $u^{-}\left(t + \frac{x}{c}\right)$ representam duas ondas progressivas com direções de propagação opostas.

Supondo a uma onda plana dada por $u(x,t) = U(x,t)e^{j\omega t}$e impondo as condições de contorno $u(x,t) = U(0,t)e^{j\omega t}$ (excitação do tubo por uma onda plana) e $p(l,t) = 0$ (ou seja, a pressão na saída do tubo, ou seja, nos lábios, é nula), sendo l o comprimento do tubo, chega-se na seguinte solução:

$$u(x,t) = \frac{cos\omega((l-x)/c)}{cos\omega l/c} U(0,\omega)e^{j\omega t}, \quad (28)$$

$$p(x,t) = j\frac{\rho c}{A}\frac{sin\omega((l-x)/c)}{cos\omega l/c} U(0,\omega)e^{j\omega t}. \quad (29)$$

O fluxo na saída do tubo é dado pela Equação 30:

$$u(l,t) = \frac{1}{cos\omega l/c} U(0,\omega)e^{j\omega t} = U(l,\omega)e^{j\omega t}. \quad (30)$$

A relação de amplitudes será dado pela Equação 31:

$$V(\omega) = \frac{U(l,\omega)}{U(0,\omega)} = \frac{1}{cos\omega l/c}. \quad (31)$$

Sendo esta relação a resposta na frequência do tubo. Para $l = 17{,}5cm$ e $c = 350m/s$ obtém-se a resposta mostrada na Figura A.21.

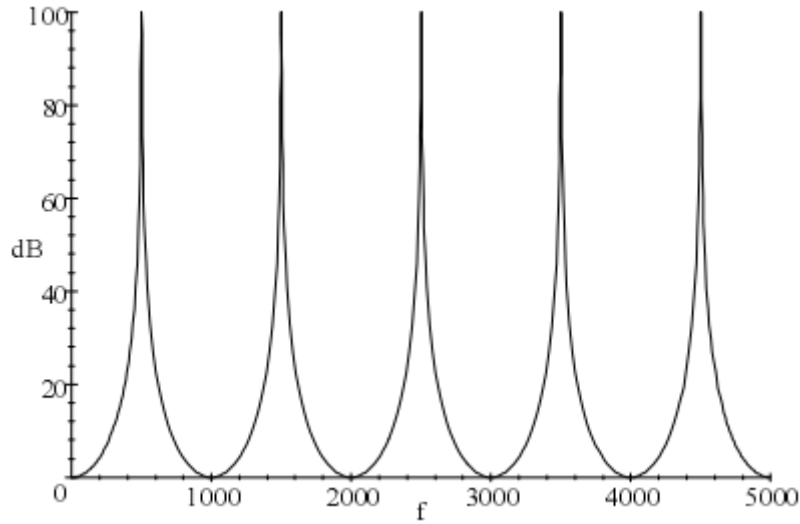



Figura A.21: curva Frequência (Hz) x Intensidade (dB). Fonte: Google Images.

Os pólos ocorrem às frequências $f_i = \frac{c}{4l}(2i - 1), i = 1,2, ...$

Um modelo muito útil consiste em considerar o trato vocal composto por uma série de tubos acústicos uniformes acoplados tal como se representa na Figura A.22.

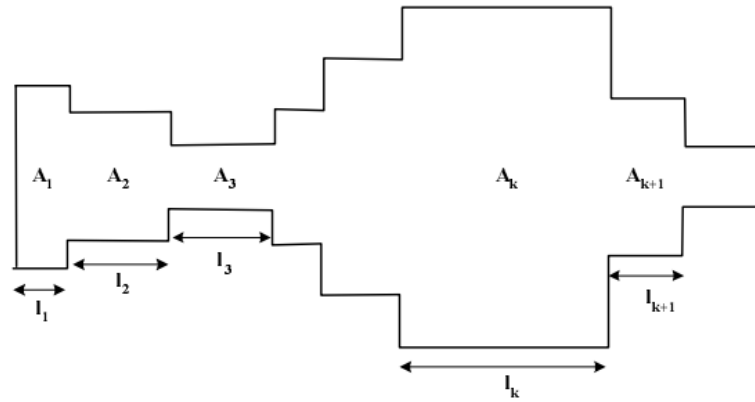



Figura A.22: modelo do trato vocal baseado em tubos de dimensões diversas. Fonte: Google Images.

Introduzindo o tempo de atraso do tubo de ordem como sendo:

$$\tau_k = \frac{l_k}{c}, \tag{32}$$

E o coeficiente de reflexão r_k na junção entre os tubos de ordem k e $k+1$:

$$r_k = \frac{A_k - A_{k+1}}{A_k + A_{k+1}}, \tag{33}$$

(note que$-1 \leq r_k \leq 1$).

Efetuando algumas manipulações matemáticas, chega-se às seguintes expressões:

$$u_{k+1}^{+}(t) = u_k^{+}(t-\tau_k)(1-r_k) - r_k u_{k+1}^{-}(t), \quad (34)$$

$$u_k^{-}(t+\tau_k) = r_k u_k^{+}(t-\tau_k) + (1+r_k) u_{k+1}^{-}(t), \quad (35)$$

Estas equações mostram que cada onda que chega à junção k se decompõe em duas, uma que é transmitida para a seção seguinte e outra que é refletida, que pode ser representado por um diagrama de fluxo de sinal mostrado na Figura A.23.

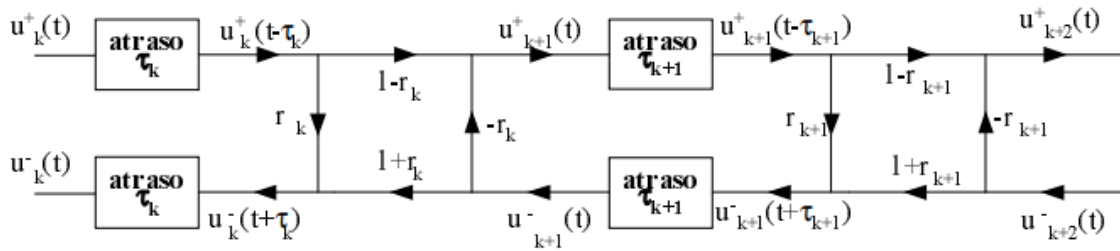



Figura A.23: diagrama de fluxo de sinais para o modelo proposto. Fonte: Google Images.

A partir de analogias entre o modelo de propagação de ondas num tubo acústico e o modelo de propagação de ondas eletromagnéticas numa linha de transmissão, pode-se estabelecer algumas relações de equivalência. Pode-se definir uma impedância acústica pela relação:

$$Z(x,\omega) = \frac{p(x,t)}{u(x,t)} = \frac{P(x,\omega)}{U(x,\omega)}. \quad (36)$$

Usando esta analogia, pode-se definir a impedância característica de um tubo uniforme e sem perdas como:

$$Z_0 = \frac{\rho c}{A}, \quad (37)$$

O papel da glote e dos lábios pode ser modelado usando tubos "semi-infinitos" sem perdas, conforme mostrado na Figura A.24. Num tubo destes uma onda aplicada à sua entrada se propagará sem reflexões:

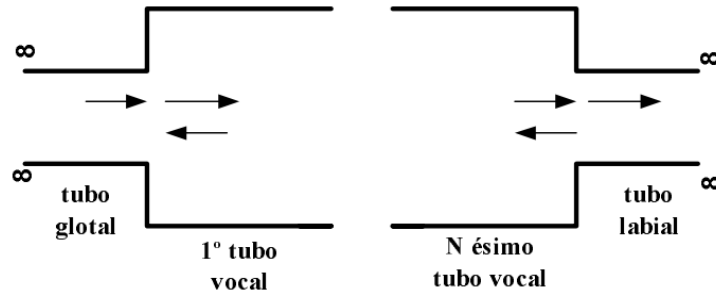



Figura A.24: modelo de tubos semi-infinitos. Fonte: Google Images.

Define-se coeficiente de reflexão labial por:

$$r_{lab} = \frac{Z_{lab} - Z_{0,N}}{Z_{lab} + Z_{0,N}}. \quad (38)$$

Para estudar o modelo da glote, pode-se recorrer ao modelo de circuito mostrado na Figura A.25:

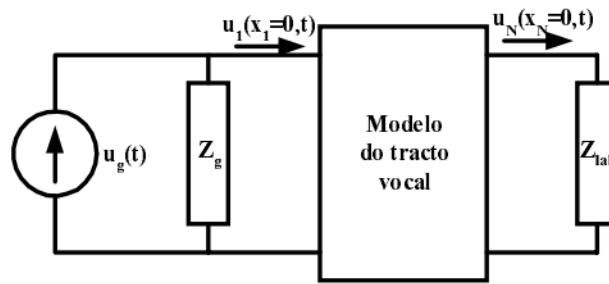



Figura A.25: modelo de circuito para a glote. Fonte: Google Images.

Fazendo uma analogia da velocidade $u(t)$ com a intensidade de corrente, tem-se:

$$u(0,t) = u_g(t) - \frac{p_1(0,t)}{Z_g}, \quad (39)$$

Considerando:

$$r_g = \frac{Z_g - Z_{0,1}}{Z_g + Z_{0,1}}, \quad (40)$$

Esses resultados nos permitem construir o diagrama de sinal mostrado na Figura A.26:

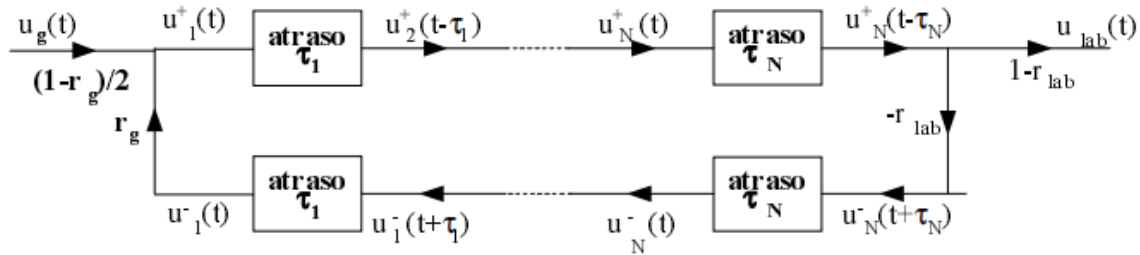



Figura A.26: diagrama de Sinais. Fonte: Google Images.

A.3.3 Modelo de Tubos: Caso Discreto

Considere o caso particular em que o trato vocal é composto por N tubos de comprimento $l_k = \frac{l}{N}$. O tempo de propagação em cada seção é igual a $\tau = \frac{l}{cN}$. O Trato vocal pode ser modelado então por um conjunto de tubos iguais cujo diagrama de fluxo de sinais neste modelo pode ser representado pelo diagrama mostrado na Figura A.27, simplificado para 3 tubos:

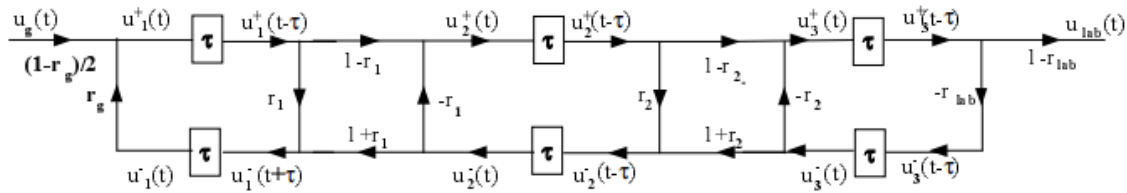



Figura A.27: diagrama de fluxo de sinais para o caso discreto. Fonte: Google Images.

Ao se aplicar um impulso unitário na entrada, o sistema responderá com um impulso após $N\tau$. Como na saída de cada tubo haverá um impulso refletido que se propagará para a entrada e será refletido novamente para a saída, novos impulsos aparecerão à saída cada 2τ. Pode-se dizer então que a resposta ao impulso do sistema é dado por:

$$u_a(t) = \sum_{k=0}^{\infty} \alpha_k \delta(t - N\tau - 2k\tau). \quad (41)$$

Uma vez que a resposta ao impulso é formada por impulsos igualmente afastados de 2τ no tempo, se aplicarmos na entrada um sinal amostrado à frequência $f_s = \frac{1}{2\tau}$, impondo obviamente que o sinal tenha sua descrição na frequência limitada a $\frac{f_s}{2}$, o sistema se comportará como um sistema digital causal com resposta ao impulso dada por:

$$u_a(n) = f(x) = \begin{cases} 0, & n < N/2 \\ \alpha_{n-N/2}, & n \geq N/2 \end{cases}. \quad (42)$$

Lembrando que um atraso de τ que é metade do período de amostragem corresponde, no domínio da transformada z à multiplicação por $z^{-1/2}$ podemos representar o sistema discreto pelo seguinte diagrama mostrado na Figura A.28:

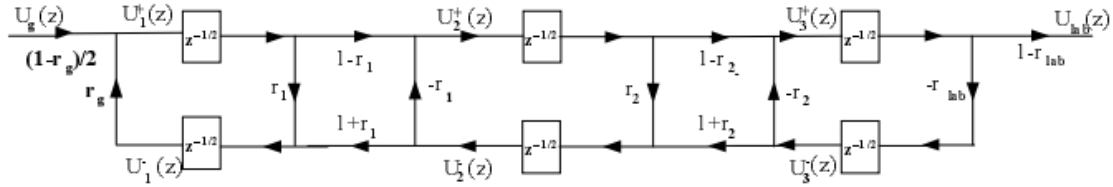



Figura A.28: diagrama de fluxo de sinais para o caso discreto. Fonte: Google Images.

Cuja função de transferência é dada por:

$$V_a(z) = \frac{U_g(z)}{U_{lab(z)}}. \tag{43}$$

É possível provar que a função de transferência é uma fração racional em potências de z^{-1}

$$V_a(z) = \frac{N(z)}{D(z)}, \tag{44}$$

Com:

$$N(z) = 0{,}5\left(1 - r_g\right)(1 - r_1) \dots (1 - r_{N-1})(1 - r_{lab})z^{-\frac{N}{2}}, \tag{45}$$

E

$$D(z) = \begin{bmatrix} 1 & r_g \end{bmatrix} \begin{bmatrix} 1 & r_1 \\ r_1 z^{-1} & z^{-1} \end{bmatrix} \dots \begin{bmatrix} 1 & r_{N-1} \\ r_{N-1} z^{-1} & z^{-1} \end{bmatrix} \begin{bmatrix} 1 & r_{lab} \\ r_{lab} z^{-1} & z^{-1} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \tag{46}$$

Que é um polinômio em z^{-1} de grau N.

Isto significa que o trato vocal pode ser representado por um sistema linear com $N/2$ zeros em $z = 0$ e N pólos:

$$V_a(z) = \frac{G_{z^{-N/2}}}{1 - \sum_{k=1}^{N} a_k z^{-k}}. \tag{47}$$

Que pode ser representado pelo diagrama de Fluxo de sinal mostrado na Figura A.29.

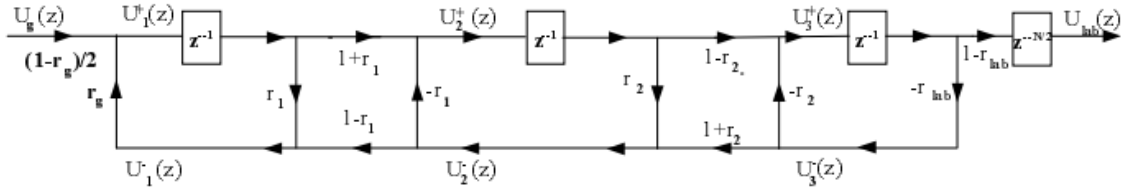



Figura A.29: diagrama de fluxo de sinais para o caso discreto. Fonte: Google Images.

Observando o fato de que zeros na origem não afetam a resposta em frequência, o modelo que se usa para o trato vocal é o modelo só com pólos ou atoregressivo:

$$H(z) = \frac{1}{A(z)} = \frac{G}{1 - \sum_{k=1}^{N} a_k z^{-k}}. \tag{48}$$

Um modelo discreto completo para a produção de voz é mostrado na Figura A.30:

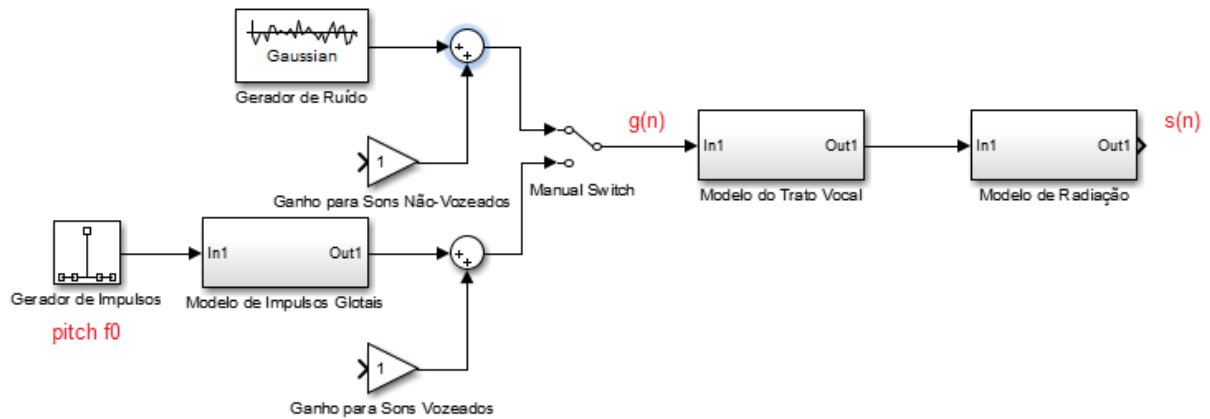



Figura A.30: modelo discreto completo para a produção de voz.

Sons vozeados são produzidos por uma excitação constituída por impulsos produzidos nas pregas vocais e sons friccionais resultam da excitação do trato vocal por um fluxo turbulento de ar. Assim, a fonte de excitação para os sons friccionais pode ser um gerador de ruído e a fonte para sons vozeados pode ser um gerador de impulsos periódicos de forma apropriada. Uma expressão muito usada é dada por:

$$g(n) = \begin{cases} 0{,}5\left(1 - \cos\left(1 - \frac{\pi n}{P}\right)\right), 0 \le n \le P \\ cos\left(\frac{\pi(n-P)}{2(K-P)}\right), P < n \le K \\ 0 \; n > K \end{cases}, \tag{49}$$

em que P é o instante do valor de pico e K o instante em que ocorre a oclusão completa.

O código desenvolvido em Matlab mostrado abaixo gera sinais glotais, cujas respostas são mostradas nas Figuras A.31 e A.32:

```
P=35; K=40;
for n=1:P+1
        g(n)=0.5*(1-cos(pi*(n-1)/P));
end
for n=P+1:K+1
        g(n)=cos(pi*(n-1-P)/2/(K-P));
end
plot((0:K),g)
fs=8000
t=0.04
f0=100;T0=fs/f0;
N=floor(fs*t)-K;
x=zeros(1,N);
for i=1:T0:N
        x(i)=1;
end
y=conv(x,g)
figure(2)
plot((0:N+K-1),y)
```

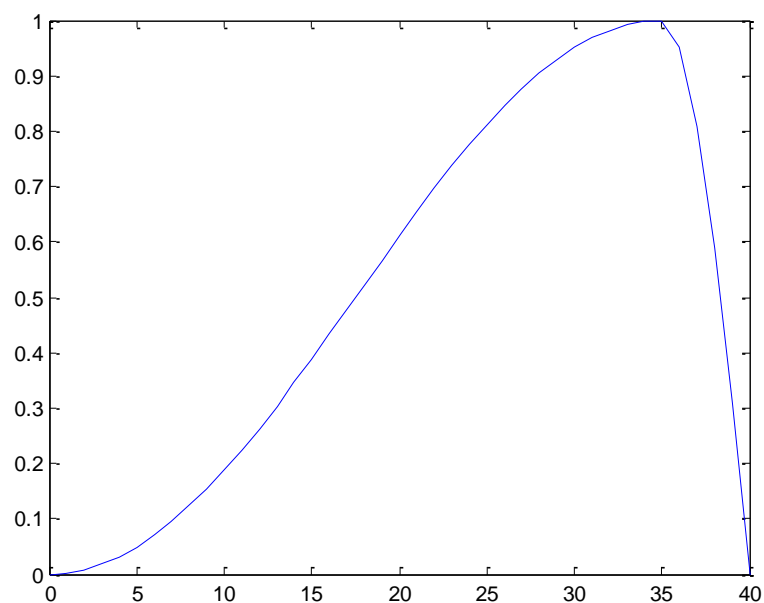



Figura A.31: resposta obtida para o código MATLAB para obtenção de sinais glotais.

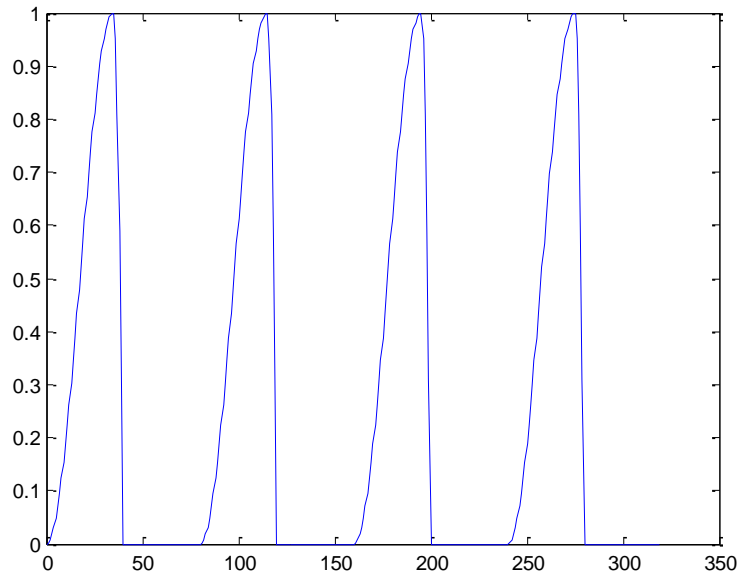



Figura A.32: resposta obtida para o código MATLAB para obtenção de sinais glotais.

A.3.4 Modelo do Trato Vocal com Perdas

O grau de resistência que o meio oferece ao movimento se traduz nos conceitos de impedância acústica, definida como o quociente entre as transformadas de Laplace da pressão e a velocidade e no conceito dual de admitância acústica, que é por sua vez o quociente entre as transformadas da velocidade a pressão. A admitância pontual é dada por:

$$Y^*(\theta, x, s) = \frac{V_n(\theta, x, s)}{P_{parede}(\theta, x, s)}, \quad (50)$$

em coordenadas cilíndricas (θ, x), sendo $P_{parede}(\theta, x, s)$ a pressão sobre a parede do trato vocal e $V_n(\theta, x, s)$ a velocidade de deslocamento da parede normal à superfície (LIMA, 2010).

Um condutor real exibe ao menos dois fenômenos: a viscosidade e a absorção nas paredes. Sendo $u(x, t)$ a velocidade volumetria e $U(x, s)$sua transformada de Laplace, então, (LIMA, 2010) mostra que $U(x, s)$ é dada pela Equação 51:

$$U(x, s) = \frac{-A}{\rho_0 s + AR}\left[a\frac{\partial g}{\partial x}(x, s) + b\frac{\partial h}{\partial x}x(x, s)\right], \quad (51)$$

em que A é a áreas, ρ_0é a densidade do ar, g e h são soluções linearmente independentes da solução geral $P(x, s) = ag(x, s) + bh(x, s)$ da Equação 52:

$$U\frac{\partial}{\partial x}\left(\frac{A(x)}{\rho_0 s + AR}\frac{\partial P}{\partial x}\right) = \left(\frac{As}{\rho_0 c^2} + Y\right)P. \quad (52)$$

Em que c é a velocidade do som no ar, P é a pressão e Y é a impedância acústica.

A.3.5 Modelo Fone-Filtro

O modelo fone-filtro é construído através dos formantes - valor nominal da frequência central da zona de ressonância em questão. Nessa zona de frequência central se encontra a maior concentração de energia (LIMA, 2010).

O modelo Fone-Filtro da produção de voz pode ser subdividido em três etapas distintas: fone, filtro (trato vocal) e a irradiação (BRANDÃO, 2011).

Seus efeitos acústicos podem ser visualizados por meio do diagrama de blocos mostrado na Figura A.33.

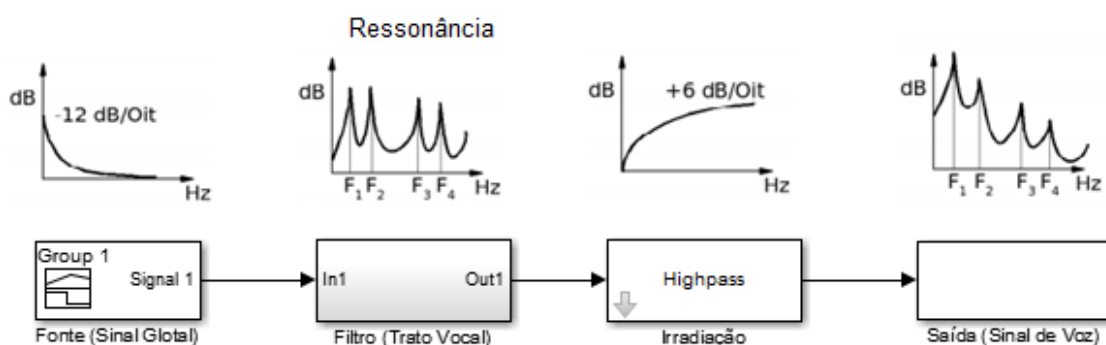



Figura A.33: diagrama de blocos para o modelofone-filtro. Fonte: (BRANDÃO, 2011 - Adaptado).

O fluxo de ar é modulado pelo movimento vibratório das pregas vocais. Graças ao "efeito Bernoulli", os pulsos de ar gerados possuem forma assimétrica, devido ao fechamento mais rápido, de modo que o sinal glotal é formado por uma série de harmônicos a ser filtrada, na etapa do trato vocal, gerando o som das vogais. Numa voz normal, a potência dos harmônicos do sinal glotal se reduz, em média, a uma taxa de 12dB por oitava. Isso gera o efeito do filtro glotal, que reduz as altas frequências. Na etapa de filtragem, a forma da estrutura do filtro (trato vocal), pode ser modificada, de modo a alterar suas formantes ou frequências de ressonância da estrutura supra-glotal. Para uma dada forma, o sinal glotal é filtrado criando o som da respectiva vogal. Na etapa de irradiação, as baixas frequências - comprimento de onda maior, sofrem difração nos lábios, enquanto as altas frequências - comprimento de onda menor, possuem maior diretividade, sendo mais suscetíveis ao efeito de reflexão. Resumindo, a etapa de irradiação amplifica as altas frequências com ganho médio de 6dB por oitava.

Já foi demonstrado que, no modelo fonte-filtro, o trato vocal pode ser considerado um sistema acústico linear. Logo, pode também ser caracterizado por uma função de resposta em frequência. O fato de ser possível obter o sinal glotal através de filtragem inversa garante que o trato vocal pode ser considerado um filtro acústico linear. Assim o modelo fonte-filtro representado no diagrama da Figura C.16 considera a linearidade do trato vocal e a inexistência da interação acústica entre o trato vocal e a fonte sonora glotal (BRANDÃO, 2011).

No modelo fonte-filtro, a sequência de amostrar s[n] é modelada como um sinal de excitação $r[n]$ aplicado por um filtro $h[n]$: $s[n] = h[n]*r[n]$. O filtro pode ser estimado a partir de um sinal de fala por meio de, por exemplo, a predição linear. A excitação, ou o resíduo $r[n]$ é encontrada por meio da filtragem inversa $r[n]=h^{-1}[n]*s[n]$. Assumindo que tal modelo é uma descrição precisa da produção de voz e que o filtro estimado apresenta comportamento muito parecido com o trato vocal verdadeiro $h^t[n]$, $r[n]$ aproxima a excitação do sinal produzido pelas pregas vocais. Consequentemente $r[n]$ é independente de $h[n]$ e a fala com um formato espectral desejado pode ser gerado aplicando $r[n]$ em um novo filtro $h'[n]$. O problema deste procedimento é que qualquer erro de estimação do filtro é atribuída à excitação. Uma vez que o resíduo passa pelo mesmo filtro, os erros compensam e o sinal de fala é reconstruído perfeitamente. Entretanto, se $r[n]$ passa por um novo filtro $h'[n]$, pequenos erros em $r[n]$ podem ser aplicados de acordo com o formato espectral do novo filtro. Desta forma, erros em regiões perceptivelmente menos importantes em regiões de $s[n]$, como em vales espectrais, podem introduzir grandes erros em importantes regiões do sinal modificado, como por exemplo, próximo às frequências formantes (WOUTERS et. al. 2000).

A.3.6 Modelo Massa-Mola

Os primeiros modelos do sistema vocal podem ser encontrados em (BRANDÃO, 2011) e representam a movimentação das pregas vocais a partir de modelos mecânicos massa-mola-amortecedor, conforme mostrado na Figura A.34.

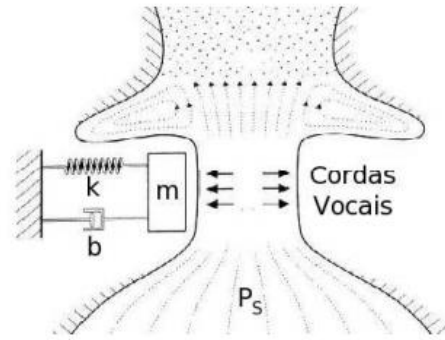



Figura A.34: modelo massa-mola-amortecedor. Fonte: (BRANDÃO, 2011).

Segundo esses modelos, as equações que fornecem a dinâmica das pregas vocais é dada pela Equação 53:

$$M\ddot{x}(t) + B\dot{x}(t) + Kx(t) = F(x,t), \qquad (53)$$

em que $x(t)$ é o deslocamento da massa M, B e K são as constantes de rigidez e elasticidade, respectivamente e $F(x,t)$ é a força aplicada ao sistema, considerada como a média entre as pressões subglotal e supraglotal.

Posteriormente, em 1972, foi proposto por Ishizaka e Flangan, um modelo para as pregas vocais considerando agora que o mesmo seria composto por duas massas. Tal modelo considerada cada uma das pregas vocais como um sistema de duas massas, ligadas às paredes da laringe por duas molas não lineares S_1 e S_2 e ligadas entre si por uma mola linear K_c, cujo modelo é esquematizado na Figura A.35 (BRANDÃO, 2011).

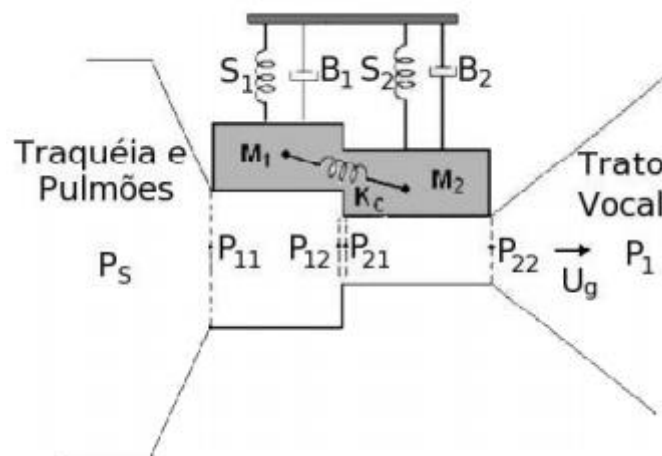



Figura A.35: modelo massa-mola com duas massas. Fonte: (BRANDÃO, 2011).

As massas movem-se somente na direção transversal. O movimento das duas pregas vocais é considerado simétrico, assim, somente é equacionado o movimento de uma

delas. Os deslocamentos $x_j(t)$ de cada uma das massas são regidos pelo sistema de equações seguinte:

$$\begin{cases} M_1\ddot{x}_1 + S_1(x_1) + B_1(\dot{x}_1) + k_c(x_1 - x_2) = F_1 \\ M_2\ddot{x}_2 + S_2(x_2) + B_2(\dot{x}_2) + k_c(x_2 - x_1) = F_2, \end{cases} \quad (54)$$

em que S_1 e S_2 são as relações das molas não lineares dadas por $S_j(x) = K_j x(1 + \eta_j x^2)$ para $j = 1,2$. Os coeficientes K_j representam a rigidez linear e η_j são coeficientes positivos que caracterizam a não linearidade das molas. As forças F_1 e F_2 dependem da pressão subglotal, do fluxo glotal e da área da região entre as pregas vocais.

Apesar dos modelos massa-mola-amortecedor serem capazes de capturar as propriedades básicas do movimento das pregas vocais, muitos detalhes desse movimento são desconsiderados devido à sua representação matemática simplificada. As pregas vocais são mais espessas na região posterior do que na região anterior, logo, sob a ação do fluxo de ar, elas se abrirão primeiro na parte anterior indo em direção à parte posterior, formando uma onda, que é a onda mucosa, a qual percorrerá a extensão das pregas vocais à medida que o fluxo de ar é mantido, como indicam as setas. Tais modelos simplificados não conseguem recriar o sistema de forma satisfatória (BRANDÃO, 2011).

(BRANDÃO, 2011) também conclui que as forças elásticas desempenham um papel importante na determinação das variáveis aerodinâmicas associadas com a qualidade vocal. Para um maior realismo, é necessário considerar sua elasticidade em cada ponto da estrutura, e não considerar o mesmo módulo de elasticidade para a estrutura inteira.

Dois problemas principais impedem a modelagem precisa das pregas vocais. O primeiro problema é relativo à sua forma exata, na qual os modelos massa-mola conseguem fazer simulações razoáveis, mas ainda não são adequados. O segundo problema é relativo à elasticidade dos tecidos, a qual varia para diferentes pontos das pregas vocais e ainda em função das contrações musculares, o que deveria ser refletido também nos modelos (BRANDÃO, 2011).

A.3.7 Modelagem Baseada em Imagens Médicas

Com as técnicas de imageamento por ressonância magnética (IRM) é possível resolver o problema da forma da estrutura na modelagem tridimensional, pois elas permitem a visualização espacial da maioria dos tecidos. Assim, é possível obter malhas individualizadas para modelagem, restando apenas o problema da determinação das

características do tecido em cada ponto, o qual pode ser resolvido através da técnica de imageamento por elastografia (BRANDÃO, 2011).

A elastografia por ressonância magnética (ERM) é uma técnica que permite obter as propriedades mecânicas dos tecidos e consiste em provocar ondas mecânicas nos tecidos e usar um equipamento de RM para medir as variações na posição dos mesmos com base nos deslocamentos observados. Assim, é possível obter imagens nas quais os *pixels* representam a elasticidade em cada ponto dos tecidos (BRANDÃO, 2011).

Certas partes do sistema de produção de voz podem ser melhor modeladas ao considerarmos o aspecto estocástico, através da modelagem de incerteza das presentes nessas partes. Isto pode ser feito através da associação de variáveis aleatórias a parâmetros do sistema e construindo, para cada variável aleatória, uma função densidade de probabilidade de acordo com uma certa estratégia. Em determinados artigos, as funções de densidade de probabilidade foram construídas com base no Princípio da Máxima Entropia, construindo um sistema dinâmico não-linear estocástico visando a geração de sons vozeados (BRANDÃO, 2011).

A tarefa da modelagem 3D das pregas vocais envolve detalhes com a colisão das pregas vocais, movimentação do fluxo de ar, variação dos carregamentos para as diferentes posições e pontos das pregas vocais, estimação/medição dos valores iniciais e de contorno (BRANDÃO, 2011).

A dependência da área de seção transversal ao longo do trato vocal é chamada Função Área do trato vocal. A função área para uma vogal, por exemplo, é determinada principalmente pela posição da língua, mas as posições do maxilar, lábios, e, em menor proporção, a do véu palatino também influenciam no som resultante (BRANDÃO, 2011).

A função área do trato vocal, mostrada na Figura A.36, fornece a área da seção transversal em relação ao eixo do trato vocal para cada ponto localizado nesse eixo a uma determinada distância da glote (BRANDÃO, 2011).

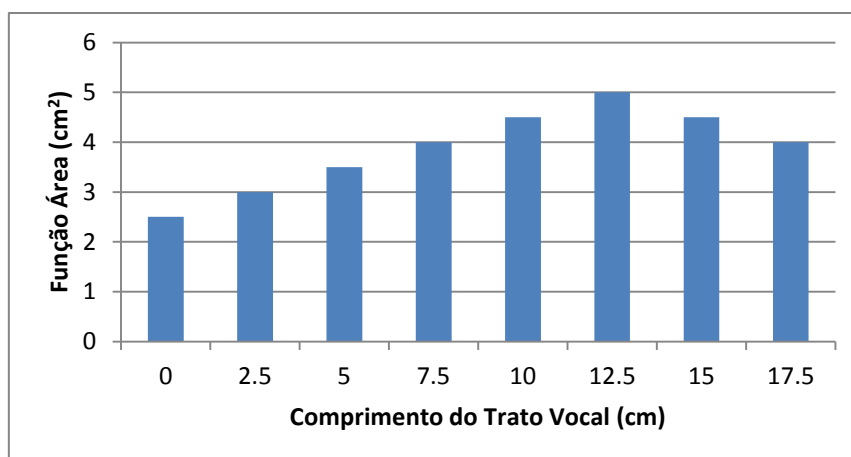



Figura A.36: função área do trato vocal. Fonte: (BRANDÃO, 2011 - Adaptado).

A função área é especificada por um número fixo de seções nos quais a k-ésima seção é definida por uma seção-transversal *A(k,n)* e comprimento *x(k,n)*. O índice *n* denota um tempo discreto. Uma transição suave entre uma a seções transversais entre um fone e outro é um item importante a ser observado (MAEDA, 1995).

A função área interpolada é alimentada no modelo do trato vocal. A variação da área vocal, geometria dos pulsos vocais parametrizados representado a oscilação quasi-periódica das pregas vocais para sons vozeados e um processo de abertura/fechamento lento da glote durante consoantes para suprimir um fluxo de ar suficiente é também calculado por meio de esquema de interpolação e usado pelo sintetizador. O ruído fricativo é automaticamente gerado pelo sintetizador. Ruído rosa é, na realidade, uma sequência de números aleatórios que passaram por um filtro pasa-baixas aplicada na seção de constrição. A amplitude do ruído é modulada por uma função da seção transversal da constrição e estimulada pelo fluxo de ar usando tanto lei quadrática como cúbica. Por meio desse modelo, é possível sintetizar vários fricativos e pausas não vozeadas em diversos contextos em conjunto com vogais (MAEDA, 1995).

Os cálculos baseados em modelos unidimensionais dependem das funções de área do trato vocal e da limitação da faixa de frequência em um certo valor. A seção transversal do trato vocal deve ser menor que a metade de um comprimento de onda para que o modelo de onda plana possa ser utilizado. A partir deste valor de frequência começam a surgir modos de propagação adicionais, não descritos pelos modelos unidimensionais. Por isso, para altas frequências, não é válido considerar a onda acústica que se propaga pelo trato vocal como uma onda plana (BRANDÃO, 2011).

Um modelo 2D apresenta precisão similar ao modelo 1D, porém, apresenta maior realismo (BRANDÃO, 2011).

A função área é importante nas simulações 1D do trato vocal e para síntese de voz. Já foram combinadas imagens de tomografia com dados acústicos e da geometria dos lábios procurando melhorar a confiabilidade na obtenção da função área (BRANDÃO, 2011).

Story criou um modelo paramétrico para controlar a função área do trato vocal permitindo a simulação de consoantes e vogais (BRANDÃO, 2011).

A dificuldade em se modelar a complexa estrutura dos órgãos do corpo humano, especialmente a laringe e o trato vocal, é que as formas aproximadas perdem detalhes como pequenas deformações, protuberâncias e assimetrias naturais dos tratos vocais reais (BRANDÃO, 2011).

A.4 O sinal de voz do ponto de vista do processamento homomórfico de sinais

Seja x um sinal de saída de um sistema linear invariante no tempo resultante da convolução de uma excitação u com sua resposta impulsional h:

$$x = u * h. \tag{55}$$

Um sinal de voz pode ser considerado como:

$$x(n) = \big(p(n) * g(n) * h_c(n) * r_l(n)\big)w(n), \tag{56}$$

em que $p(n)$ é, para sons vozeados, um trem de impulsos periódicos de período $P = \frac{1}{f_0}$:

$$p(n) = \sum_k \delta(n - kP), \tag{57}$$

Em que $g(n)$ é uma onda glotal de duração finita composta por duas partes: uma de mínima fase $g_1(n)$ e outra de máxima fase $g_2(n)$, sendo $g(n) = g_1(n) + g_2(n)$; $h_c(n)$ é a resposta impulsional do trato vocal, excetuando os sons nasais. O trato vocal é bem representado por um modelo de fase mínima só com polos; $r_l(n)$ é a resposta impulsional que traduz a radiação nos lábios e cujo efeito de radiação pode ser representado por um sistema com um zero. $R_l(z) \cong 1 + z^{-1}$ e $w(n)$ é uma janela temporal. Considerando que:

$$h(n) = g(n) * h_c(n) * r_l(n), \tag{58}$$

Então:

$$x(n) = \big(p(n) * h(n)\big)w(n), \tag{59}$$

Como $x(n)$ não é uma convolução - e para poder efetuar a desconvolução seria necessário que o fosse, pode-se tomar janelas suficientemente grandes, de dimensão M, cobrindo um número significativo de períodos do fundamental P, o que fornece a seguinte aproximação:

$$x(n) = \big(p(n)w(n)\big) * h(n) = p_w(n) * h(n), \tag{60}$$

Fazendo:

$$p_w(n) = p(n)w(n) = \sum_{k=0}^{M-1} w(kP)\delta(n - kP), \tag{61}$$

E:

$$P\left(e^{j2\pi f}\right) = W\left(e^{j2\pi Pf}\right), \tag{62}$$

$$\widehat{P_w}(n) = \widehat{w}\left(\frac{n}{P}\right). \tag{63}$$

O processamento homomórfico se baseia no cálculo do logaritmo de uma transformada do sinal. Ao se considerar um sinal amostrado, temos então o logaritmo da transformada de z:

$$\hat{X}(z) = lnX(z). \tag{64}$$

Define-se então cepstro complexo do sinal x como a transformada inversa de z de $\hat{X}(z)$:

$$\hat{X}(z) = \sum_{i} \widehat{x_n} z^{-i}. \tag{65}$$

Esta operação não linear será chamada de H. Nestas condições tem-se

$$\hat{x} = \hat{u} + \hat{h}. \tag{66}$$

O cepstro real de um sinal é a transformada inversa de Fourier do logaritmo de sua transformada de Fourier. Para sinais amostrados numa janela de duração finita $[0, N - 1]$:

$$X_k = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-jkn2\pi/N}, \tag{67}$$

$$c_n = \sum_{k=0}^{N-1} ln|X_k| e^{jkn2\pi/N}. \tag{68}$$

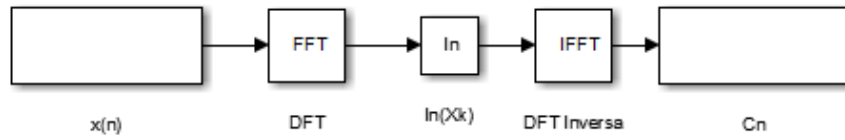



Figura A.37: análise cepstral.

O cepstro, a transformada inversa do logaritmo do espectro, complexo vale:

$$\hat{x}(n) = \widehat{p_w}(n) + \hat{h}(n). \tag{69}$$

O cepstro é composto por uma parte correspondente à resposta impulsional do sistema acústico que conta com as contribuições causais de $g_1(n)$, $h_c(n)$ e $r_l(n)$ e a contribuição não causal de $g_2(n)$, além de outra parte resultante da excitação modificada pela janela e constituída por um trem de impulsos espaçados de P amostras. O cepstro apresenta três regiões:

$-P < m < 0$: componente não causal devido a $g_2(n)$;

$0 < n < P$: componente causal devido ao trato e $g_1(n)$;

$n \cong kP$: componente devido à excitação periódica $p_w(n)$.

As duas primeiras componentes decrescem muito mais depressa que a última, o que significa que a aplicação de uma janela no cepstro permite separar as duas contribuições, e ao se calcular o cepstro inverso, pode-se determinar $p(n)$ e $h(n)$:

O cepstro é então dado por:

$$\hat{x}(n) = ln\left|X(e^{j2\pi f})\right| = ln|X| + j\beta(2\pi f), \tag{70}$$

sendo então necessário conhecer a fase de $X(e^{j2\pi f})$. As implementações de FFT normalmente apenas fornecem a parte principal da fase: $|\beta(2\pi f)| \leq \pi$, pois é necessário efetuar a operação de desenrolamento de fase (*phase unwraping*), que consiste em somar $\pm 2\pi$ nos pontos de descontinuidade. Na prática, usa-se apenas o cepstro real.

A análise cepstral é frequentemente usada em análise e processamento de sinais de voz por que é capaz de separar as características da excitação do trato vocal. Baixas frequências no cepstro representam características do trato vocal, enquanto que altas frequências representam a excitação (JUNG, 2001).

APÊNDICE B: ALGORITMOS DE SÍNTESE DE VOZ

Em 1779, o cientista dinamarquês Christian Kratzenstein, trabalhando para a Academia Russa de Ciências, desenvolveu modelos do trato vocal humano para produzir as vogais, sendo posteriormente desenvolvidos sistemas mecânico-acústicos que modelavam língua e lábios capazes de reproduzir também consoantes. Em 1930, o Bell Labs desenvolveu o vocoder. Durante os anos de 1980 e 1990, o sistema MITalk, baseado no trabalho de Dennis Klatt, no MIT e o sistema da Bell Labs foram um dos sistemas multilíngues independentes de línguas que se tornaram referências na época, usando técnicas de processamento da linguagem natural.

Os métodos de síntese de fala podem ser classificados em: concatenação de forma de onda, usando unidades sonoras, técnicas baseadas em parâmetros, síntese de formantes, síntese articulatória e síntese HMM. Tal classificação pode ser vista na Figura B.1. Todas as técnicas apresentam suas respectivas vantagens e desvantagens. De todos os três, a concatenação de forma de onda tem apresentado a maior naturalidade e seu algoritmo é bastante simples, entretanto, ainda apresenta problemas de coarticulação (KANG et. Al. 2009; TALAFOVÁ et. al., 2007).

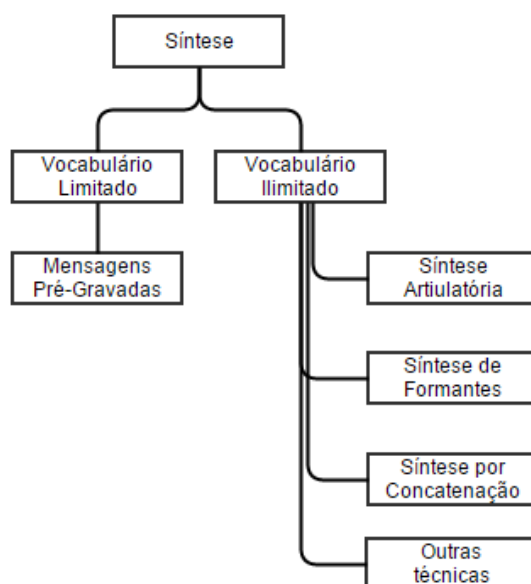



Figura B.1: classificação e aplicação dos tipos de sistemas de síntese de voz. Fonte: (AZUIRSON, 2009 - Adaptado).

Dentre as técnicas citadas, três são as principais: síntese de formantes, síntese articulatória e síntese concatenativa. A síntese de formantes modela as frequências do

sinal de voz. Formantes são as frequências de ressonância do trato vocal. A síntese é realizada usando tais frequências estimadas. A síntese articulatória gera voz a partir de modelos do comportamento articulatório do trato vocal humano. A síntese concatenativa produz fala por meio da concatenação de pequenas unidades de fala gravadas previamente, sejam fonemas, dífonos ou trífonos. A síntese por seleção de unidades ao invés de gravar apenas uma unidade sonora, grava diversas - até centenas, de ocorrências da mesma unidade (TABET, 2011).

A forma mais simples de um sistema TTS é utilizar um "*Look-up Table*", simplesmente reproduzindo vozes pré-gravadas e codificadas. Tal abordagem é utilizada em aplicações com poucas mensagens (SHAUGHNESSY, 2003).

Métodos de síntese baseados em manipulação do espectro do sinal de voz - como síntese de formantes ou síntese por codificação preditiva linear, produzem uma saída compreensível, porém pouco natural. Essa falta de naturalidade provém de modelos muito simplificados, inventários com poucas unidades sonoras ou controle de prosódia ruim (KOBAYASHI et. al.).

Para o português brasileiro, as técnicas mais empregadas são a síntese concatenativa e a síntese baseada em formantes (COSTA e MONTE, 2012).

Em sistemas que exigem apenas um vocabulário limitado, baseados em palavras ou frases previamente gravadas, é possível gerar pequenas frases com alta naturalidade e inteligibilidade, uma vez que é possível gravar todas as palavras ou trechos necessários para síntese em seus mais variados contextos (AZUIRSON, 2009).

Para sistemas limitados, a síntese paramétrica é bastante adequada: a síntese paramétrica possui um banco de palavras parametrizadas utilizando técnicas de parametrização , coeficientes LPC, sendo possível a recuperação do sinal original sem perda considerável de qualidade. Uma de suas vantagens é a redução do espaço de armazenamento requerido, uma vez que o que se armazena é a palavra parametrizada e não a forma de onda propriamente dita. Além disso, ao se manipular os parâmetros, é possível obter transições mais suaves, tornando a saída mais natural. Assim, sistemas de síntese em domínio específico apresentam alta naturalidade devido ao fato das sentenças serem limitadas, não apresentando propósito de uso geral, apenas para funções pré-programadas (AZUIRSON, 2009).

A síntese de vocabulário ilimitado tenta obter todas as informações sobre prosódia necessária para a síntese a partir do texto. São portanto, mais abrangentes. O espaço de

armazenamento exigido também é menor, por armazenarem menos informações - como unidades sonoras menores (AZUIRSON, 2009).

Vozes sintetizadas de alta qualidade podem ser construídos a partir de um banco de dados diversificado de obtido a partir de uma voz natural de um único locutor. Os inventários, comumente encontrados em sistemas baseados em dífonos, ficaram mais generalizados e portanto, tem consumido mais recursos. Por este motivo, estudos tem procurado formas de selecionar automaticamente unidades sonoras a partir de grandes bancos de dados de vozes naturais, se tornando uma técnica de síntese dominante, criando soluções baseadas em esquemas de treinamento e com aplicações em diversas línguas. Tal estratégia tem levado os sistemas comerciais a um outro nível. Embora o cenário seja bastante promissor, as técnicas de síntese de voz ainda apresentam falhas. É impossível garantir que não haja concatenações ruins ou seleção inapropriada de unidades sonoras devido ao grande número de combinações possíveis existentes. Entretanto, para determinadas aplicações - limitadas a aplicações específicas, é possível quase sempre, evitar falhas (BLACK, 2007).

Atualmente, seleção de unidades e a concatenação são uma das formas de síntese de voz mais usadas no mundo, tanto para aplicações acadêmicas como comerciais. Algumas dessas técnicas não realizam modificações na prosódia, enquanto outras geram forma de onda modificando os contornos da frequência fundamental e a duração das unidades selecionadas de acordo com a prosódia.

Os dois sistemas de síntese mais bem sucedidos atualmente são a síntese concatenativa (CSS - *Concatenative Speech Syntehsis*) e o baseado em Modelo de Markov Oculto (HMMSS - *Hidden Markov Model based Speech System*). O CSS é baseado na concatenação de segmentos de voz gravados. Nas primeiras versões de sistemas CSS simples, unidades eram armazenadas para reuso para síntese em contextos diferentes sem algum tipo de modificação, o que gerava resultados com pouca qualidade devido a problemas de contexto. Atualmente, usa-se um amplo banco de dados cobrindo todas as possibilidades possíveis e selecionando a unidade mais adequada para concatenação. Entretanto, o uso de grandes bancos de dados tornam a seleção de unidades convencional impossível de serem usados em dispositivos com espaço para armazenamento limitado ou em condições com dados limitados (PHUNG et. al.).

HMMSS representa um modelo baseado no domínio estatístico, ao invés de forma de onda ou espectral. O tamanho dos parâmetros estatísticos treinados são pequenos, o que possibilita que sistemas HMMSS sejam distribuídos para diferentes plataformas.

Modelagem de contexto relacionado a coarticulação também é bem realizada com HMMSS, resultando em uma saída suave. Entretanto, para garantir uma boa precisão estatística, modelos HMM geralmente exigem uma grande quantidade de dados para treinamento (PHUNG et. al.).

B.1 Síntese articulatória

Se baseiam em técnicas computacionais de modelar o trato vocal humano e o processo articulatório que nele ocorre. O primeiro sintetizador articulatório, denominado ASY, foi desenvolvido na metade dos anos de 1970 por Phillip Rubim, Tom Baer e Paul Mermelstein nos Jaskins Laboratories. Este sintetizador foi baseado nos modelos de trato vocal desenvolvido pela Bell Labs nos anos de 1960 e 1970 por Paul Mermelstein e Cecil Coker. O sistema mais notório desenvolvido foi um sistema baseado no NeXT da Trillium Sound Research, uma compania originada na Universidade de Calgary e publicado com licença GNU e usava um modelo baseado em guias de onda e linhas de transmissão analógicas dos controles dos tratos vocal e nasal.

A síntese articulatória gera fala a partir da modelagem direta do comportamento do sistema articulatório humano, usando modelos computacionais dos articuladores (língua, lábios, etc.) e glote para sintetizar voz. Ao invés de descrever o sinal propriamente dito, a síntese articulatória emprega parâmetros de controle como posição e movimento das línguas, abertura glotal e outros parâmetros significantes para a produção de voz, assim, a síntese articulatória tenta simular o aparelho fonador humano e mimetizar a dinâmica dos articuladores (língua, mandíbula, lábios, osso hióide, véu palatino, etc.), objetivando construir o modelo mais realista possível a fim de se obter uma fala exatamente igual à humana. Matematicamente, a síntese articulatória pode ser tão simples quanto descrever o trato vocal como tubos de seção transversal variável ou tão complicado quanto resolver equações de Navier-Stokes (TABET, 2011; AZUIRSON, 2009; SCHROETER, 2005).

Uma síntese articulatória altamente precisa, teoricamente, seria capaz de produzir uma síntese completamente configurável, capaz de produzir sons de diversos locutores, estilos de fala, etc., levando em conta os limites fisiológicos da movimentação dos articuladores, bem como a interação na movimentação dos articuladores entre si (SCHROETER, 2005; AZUIRSON, 2009).

Porém, há duas grandes dificuldades nisso: aquisição de dados para modelo e o equilíbrio entre precisão/qualidade e facilidade de implementação e controle. Os dados

para o modelo geralmente são obtidos por meio de imagens de Raio-X e não caracterizam massa nem graus de liberdade (TABET, 2011).

Uma forma elegante de gerar voz seria a síntese articulatória, que em essência, transforma entradas de texto em comandos musculares a fim de criar uma sequência temporal de formatos do trato vocal, que são convertidos em filtros digitais e excitados sejam por ruídos ou pulsos periódicos (SHAUGHNESSY, 2003).

Teoricamente seria o modelo que mais deveria atingir a qualidade em seus resultados. Entretanto, na prática, é um dos métodos mais difíceis de serem implementados, devendo controlar parâmetros como abertura e formato dos lábios, posição das línguas e suas dimensões. Ademais, tal modelo é de grande complexidade computacional e nunca produziu resultados com boa qualidade, sendo, em geral, inferiores aos obtidos por meio da síntese de formantes ou síntese concatenativa (TABET, 2011; AZUIRSON, 2009; SHAUGHNESSY, 2003).

B.2 Síntese de formantes (ou síntese baseada em regras)

Para maior parte da história da síntese de voz (1965-1995), a abordagem usando filtro de envelope espectral orientados a excitação era a técnica dominante.

A síntese por formantes não usa qualquer amostra de voz humana, mas apenas em regras definidas por linguistas para gerar os parâmetros e as transições de um fonema para outro (coarticulação). Tais regras são resultado de profunda análise e estudo de espectogramas e da evolução dos formantes realizados por linguistas. Assim, este método é por vezes chamado de síntese baseada em regras. Porém, ainda não se conhece uma regra ótima (TABET, 2011).

O modelo de formantes é baseado também no modelo fone-filtro, sendo necessário, modelar a fonte de excitação (determinando seus parâmetros como amplitude, presença/ausência de ruído durante aspiração e período) e os filtros capazes de simular o trato vocal (e sua configuração como frequência, amplitude, largura de banda dos formantes e presença de zeros e polos nasais) por meio de funções de transferência. A síntese de formantes é usada em sistemas como MITalk, KlatTalk e DECTalk (TABET, 2011).

Sua vantagem reside no fato de utilizar uma representação mais econômica, exigindo pouca memória, uma vez que armazena apenas um conjunto de parâmetros juntamente com conjunto de regras de transcrição, o que torna tal solução interessante

para sistemas embarcados e sistemas com recursos de memória limitada em geral, pois não faz uso de grandes bancos de dados com amostras de voz (TABET, 2011).

Outra vantagem desta técnica é que os parâmetros estão altamente correlacionados com a produção e propagação de som no trato vocal, assim, apresenta grande flexibilidade quanto ao tipo e a qualidade das vozes geradas por meio de mudança nas regras ou nos valores para os parâmetros (TABET, 2011; AZUIRSON, 2009).

A síntese baseada em regras é, também, bastante inteligível mesmo quando o resultado é reproduzido em alta velocidade (TABET, 2011; SCHROETER, 2005).

Parâmetros como frequência fundamental, nível de ruídos são variados ao longo do tempo para gerar formas de onda. A maioria destes sistemas geram vozes muito artificiais, robóticas não atingindo naturalidade. Entretanto, a máxima naturalidade nem sempre é um objetivo primário dependendo do sistema. E a síntese de formantes pode apresentar certas vantagens sobre sistemas como os concatenativos. A síntese de formantes é inteligível mesmo em altas velocidades. Além disso, costumam ser programas menores que aqueles baseados em concatenação por não precisarem de um banco de dados de amostras, podendo ser usados em sistemas embarcados com recursos de memória e processamento limitados (TABET, 2011).

É muito mais fácil modificar os parâmetros em síntese de formantes para simular diversas vozes sintéticas que em outras técnicas, mas infelizmente é mais difícil a obter e determinar parâmetros adequados, sendo necessário um estudo do espectro da fala natural, uma tarefa muitas vezes difícil tanto em trechos estáveis como transitórios da fala (SHAUGHNESSY, 2003; AZUIRSON, 2009).

Determinar com precisão os momentos de fechamento glotal (o fechamento da prega vocal causa maior excitação do trato vocal e define o início de um período de *pitch*) também é difícil. Assim, encontrar regras para sintetizar voz é o principal problema na síntese de formantes. As regras para especificar os *timings* da voz (vozeados / não-vozeados) e os valores dinâmicos de todos os parâmetros dos filtros é também uma tarefa difícil de fazer manualmente, até mesmo para palavras simples. A obtenção dessas regras pode ser feito po meio de análise-por-síntese. Da mesma forma, técnicas automáticas para especificar os parâmetros formantes ainda não apresentam bons resultados, devendo, muitos deles serem otimizados manualmente (SHAUGHNESSY, 2003; SCHROETER, 2005; TABET, 2011).

Ademais, a síntese de formantes requer esforço computacional moderado (SCHROETER, 2005).

Na síntese por formantes, assume-se que a função transferência do trato vocal pode ser satisfatoriamente modelada por meio de simulação das frequências e amplitudes formantes, ou seja, a síntese consiste em por meio da reconstrução artificial das características formantes a serem produzidas, o que é feito por meio da excitação de ressonadores por meio de uma fonte vozeada ou gerador de ruído a fim de se obter o espectro desejado, controlando a fonte de excitação, simulando sons vozeados ou não vozeados. A adição de um conjunto de anti-ressonadores permite também a simulação de efeitos do trato nasal, fricativos e pulsantes. A especificação de 20 parâmetros resulta em um sinal de fala satisfatório (TABET, 2011).

Um conjunto de parâmetros caracterizando um envelope espectral em um curto espaço de tempo é armazenado para cada número de unidades sonoras. Uma excitação simplificada é convoluída com a resposta ao impulso de um filtro. A fonte de excitação pode ser um trem de pulsos periódicos vocais, simulando a vibração glotal, ou ruído branco, simulando sons fricativos resultantes da constrição do trato vocal ou aspirativos ou ainda ambos, ou seja, a excitação periódica é geralmente um trem de pulsos periódicos para simular sons vozeados e ruído pseudo aleatório para sons não vozeados (SCHROETER, 2005).

Enquanto que pulsos glotais para excitações vozeadas decrescem em intensidade com a frequência, a excitação de ruído para sons não vozeados é melhor modelado por um espectro plano. As intensidades de um ruído natural se aproximam da distribuição gaussiana. Amostras de ruído de excitação geralmente se originam de um gerado e de números pseudo aleatórios que levam a um espectro contínuo em distribuição uniforme. Entretanto, ao se somar diversos números aleatórios nos aproximamos de uma amostra de ruído gaussiano via teorema do limite central da probabilidade (SHAUGHNESSY, 2003).

Síntese de formantes emprega seções de filtros de segunda ordem em cascata (série) ou em paralelo. O sistema é composto pela função de transferência do trato vocal que relaciona o fluxo de volume de ar nos lábios (saída) e o fluxo do volume de ar na glote (entrada). A tarefa é aproximar todas as ressonâncias do trato vocal (picos na função de transferência, os formantes) por uma rede de filtros de segunda ordem (SCHROETER, 2005).

Pode ser demonstrado que a representação por filtros em série aproxima razoavelmente bem o trato vocal não nasal. Nesta abordagem, especificamos apenas as

frequências dos formantes, a largura de manda e o fator de ganho (SCHROETER, 2005).

Tipicamente, o filtro é especificado em termos de frequência central de ressonância (formante) e largura debanda, sobre um intervalo de frequência de aproximadamente 5 kHz.

Na síntese baseada em formantes, as quatro frequências centrais mais baixas dos formantes variam dinamicamente de frame-a-frame juntamente com as três menores bandas. Os parâmetros de ordem mais elevada são geralmente mantidos fixos, uma vez que sua variação apresenta muito pouco efeito percentual.

A abordagem clássica proposta por Klatt envolve tanto estruturas de filtros de segunda ordem em cascata e paralelos, cada um simulando uma ressonância. Esta abordagem propõe estrutura em cascata - para um fácil controle dos sons vozeados, e em paralelo para os sons fricativos. A estrutura em cascata é melhor para sons vozeados, aproximando seu envelope espectral e permitindo um único controle de amplitude. Ressonadores digitais de segunda ordem são usados em síntese de formantes porque filtros de ordem superior requerem bits adicionais nos seus coeficientes multiplicadores para atingir a mesma precisão espectral.

Os fonemas vozeados que dominam a fala, tanto em tempo como energia, são excitados na glote, assim, o filtro modela todo o trato vocal.

As obstrutivas excitam apenas uma pequena porção do trato vocal, gerando ruídos obstrutores. Assim, para os obstrutores, o trato vocal fornece ressonâncias de frequência maior e bem menos energia nas baixas frequências. Usar ressonadores em cascata é inadequado em tais circunstâncias, uma vez que os parâmetros devem mudar abruptamente ao se mudar de sons vozeados para obstrutores, sendo mais conveniente usar filtros de segunda ordem em paralelo com os mesmos parâmetros dos filtros em cascata, excetuando o controle de amplitude. Enquanto que para os sistemas em série apresentam um único controle de amplitude, na estrutura em paralela é variada separadamente.

Um banco de filtros paralelos pode ser usado para sons vozeados, mas cada amplitude formante deve ser especificada individualmente. Tal controle de amplitude é essencial para os obstrutores, porém desnecessário para os vozeados. Filtros paralelos criam flexibilidade para aproximar qualquer espectro, mas requerem ganhos individuais, além de frequências de formantes e larguras de banda. Uma outra desvantagem da abordagem unicamente paralela é a ocorrência não intencional de zeros espectrais entre

as frequências formantes, mas que podem ser canceladas por meio de filtros de correção especiais (SCHROETER, 2005).

Entretanto, para sons nasais bem como sons fricativos, a representação por filtros de segunda ordem pode não ser boa o suficiente. Sons nasais apresentam estrutura de formantes similares, um formante por quilohertz em média para um homem adulto. Quando o trato nasal é envolvido, porém, o trato naso-vocal é maior e apresenta uma ou duas ressonâncias a mais, além de zeros espectrais, sendo usados então cinco ressonadores de segunda ordem em cascata com um ressonador extra e um anti-ressonador em cascata. Nasais velares apresentam mais de um zero espectral, mas geralmente é modelado apenas um, uma vez que os outros zeros adicionais apresentam pouca importância percentual (SCHROETER, 2005; SHAUGHNESSY, 2003).

A Figura A.2 mostra o diagrama de blocos de um sistema genérico baseado em síntese de formantes, exibindo o caso para filtros em série e para filtros em paralelo.

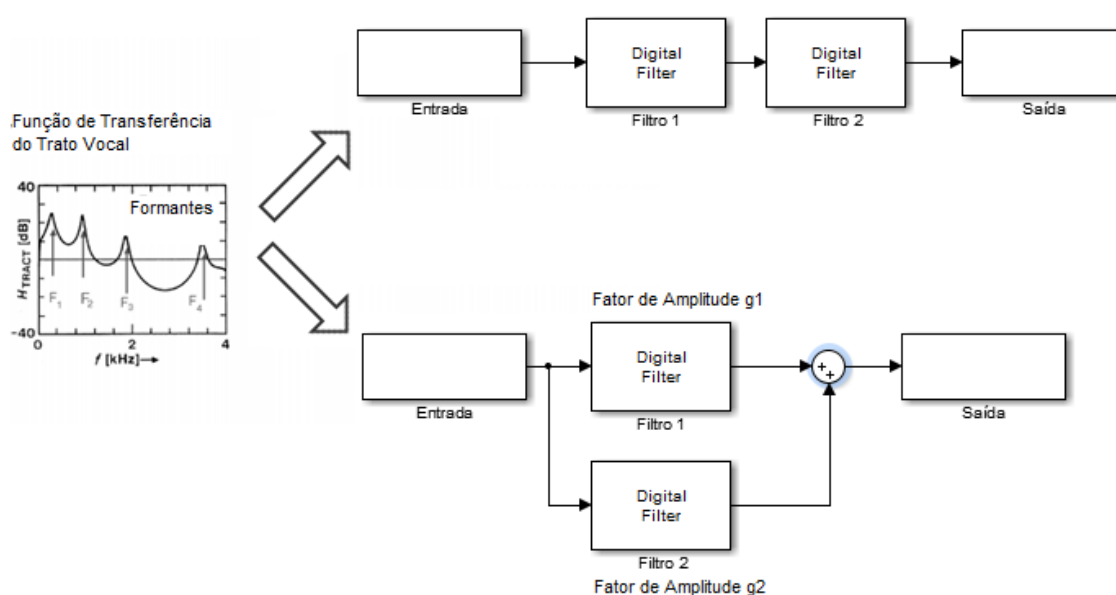



Figura B.2: diagrama de blocos explicando a síntese baseada em formantes. Fonte: (SCHROETER, 2005 – Adaptado e Traduzido).

B.3 Síntese baseada em seleção automática de unidades

Na síntese concatenativa, os dífonos devem ser modificados por meio de técnicas de processamento de sinais a fim de produzir a prosódia desejada. Tal modificação pode resultar em sons não naturais. A síntese de seleção de unidades resolve este problema armazenando no inventário múltiplas instâncias de cada unidade sonora, com diversas prosódias. A unidade se se adequa mais é prosódia é então selecionada por um

algoritmo de seleção baseado na minimização de funções custo, chamadas de função-alvo e função de concatenação, e concatenada de tal forma que as modificações nas unidades sejam mínimas ou simplesmente não sejam necessárias (TABET, 2011).

A síntese por seleção unidades fornece grande naturalidade, pois há pouca necessidade de grandes alterações na unidade gravada por meio de técnicas de processamento digital de sinal, que tornam a voz menos natural, exceto, eventualmente em alguns pontos de concatenação, além de minimizar a descontinuidade espectral e prosódica. Assim, a seleção automática de unidades requer muito menos modificação das unidades sonoras, o que leva a uma qualidade geral maior e mais natural quando comparada com síntese baseada em dífonos. Apesar disso, a síntese por seleção de unidades apresenta uma série de desvantagens como custo e tempo de desenvolvimento para coletar e rotular dados (TABET, 2011).

Para se atingir a máxima naturalidade, indistinguível da voz humana, são necessários, porém, bancos de dados da ordem de gigabytes de dados pré-gravados, o que significa dúzias de horas de gravação. Recentemente, pesquisadores tem focado em métodos automáticos para detectar segmentos não-naturais durante a seleção das unidades.

Durante a criação do banco de dados, são gravadas uma das unidades: fonemas, dífonos, semi-fonemas, sílabas, morfemas, palavras, frases e sentenças. A divisão em segmentos é feita usando reconhecedores de palavras aplicados em representações visuais como formas de onda ou espectogramas e um índice das unidades no banco de dados é criado baseado na segmentação e em parâmetros acústicos como frequência fundamental, duração, posição na sílaba e fonemas vizinhos. Em tempo de execução, a unidade é escolhida determinando o melhor candidato. Tal escolha geralmente é feita usando uma árvore de decisão.

A seleção de unidades usa grandes bancos de dados com vozes pré-gravadas. No caso de uma seleção automática de unidade, a influência coarticulatória não é limitada ao último fonema. O banco de dados é muito maior, com duração variando de 1 a 10 horas, e contempla várias ocorrências de cada unidade sonora, capturada em vários contextos, como fonemas vizinhos diferentes, *pitch*, duração, posição na sílaba, etc. (TABET, 2011).

A disponibilidade de boas ferramentas de rotulação automática de voz e a disponibilidade de diversas instâncias de um tipo de unidade específico (com diferenças de *pitch*, duração, contexto, linguístico) permitiu que a síntese por seleção de unidade se

tornasse uma solução viável. Tal método permite que se use grandes bancos de dados de voz gravados usando estilos de fala específicos e cuidadosamente controlados, como felicidade, alegria, etc. Além de, evidentemente, poder ser usado com banco de dados pequenos para aplicações específicas. Para aplicações gerais, como ler e-mails e notícias, é exigido em geral 10h de gravações a fim de se atingir uma qualidade desejável e várias dúzias para se obter uma gravação "natural". Ao contrário da síntese concatenativa, a seleção automática de unidades seleciona as unidades de síntese ótimas a partir de um inventário que pode conter uma diversidade de *tokens* de uma unidade específica a fim de concatenar para produzir a síntese. Tal técnica tem se mostrado bem sucedida. O processo de seleção da sequência ótima é automatizada por meio de *search queries* nas *strings* das *tags* do fonemas (SCHROETER, 2005).

Alguns trabalhos tem preferido o uso de semi-fonemas ao invés de dífonos, uma vez que semi-fonemas permitem que o algoritmo de busca criem dífonos que não foram gravados a partir dos semi-fonemas. A busca ótima pelas unidades é dependente de fatores como a similaridade espectral nos contornos das unidades e rótulos prosódicos configurados pelo *front-end* (SCHROETER, 2005).

Durante o treinamento, as unidades são escolhidas para o banco de dados em função de dois custos a serem minimizados: o da escolha apropriada da unidade durante a execução para um dado contexto fonético e a junção bem sucedida das unidades. Caso as unidades armazenadas sejam de baixa qualidade ou redundantes, o resultado será ruim. Caso as unidades sejam boas, mas as transições ruins, a síntese apresentará muitas descontinuidades. O algoritmo deve examinar muitas frequências diferentes dentre as unidades elegíveis existentes no banco de dados, calculando os custos propostos para cada unidade em termos de custo calculado para cada característica desejada da unidade e o custo de concatenação. Caso a distorção seja excessiva, o banco de dados é atualizado, adicionando novas unidades ou atualizando o banco de dados, reduzindo a distorção média. Tal solução pode ser aplicada também a distorções intersegmentais e a falhas acústicas causadas pela suavização requerida pelas unidades na função de quadros temporais adjacentes. Embora se deseje minimizar todos os custos, ainda não há resultados claros sobre qual deve predominar.

Um exemplo de treinamento é a “clusterização” de árvore de decisão, na qual unidades de contextos fonéticos são escolhidas por seus efeitos similares nos parâmetros acústicos ou fonemas individuais. Árvores de decisão são construídas sem intervenção humana para maximizar a similaridade acústica dentre as classes selecionadas. Apenas

um pequeno subconjunto do espaço de busca teórica é de fato usada Assim, sistemas tendem a sintetizar uma ampla quantidade de falas durante o treinamento a fim de descobrir quais unidades e junções são mais adequadas geralmente usando técnicas de programação dinâmica (SHAUGHNESSY, 2003).

B.3.1 O trabalho de (HUNT, 1996)

O uso de um banco de dados com uma grande quantidade de unidades disponível com prosódia e características espectrais diversificadas permite que se sintetize uma voz mais natural que podem ser produzidas por meio de uma pequena quantidade de unidades controladas (HUNT, 1996).

O primeiro estágio é transformar a entrada em especificação-alvo: os fonemas solicitados em conjunto com características prosódicas, como *pitch*, duração e potência (HUNT, 1996).

A seleção de unidades é baseada em duas funções custo: custo-alvo $C^t(u_i, t_i)$, a estimativa da diferença entre a unidade do banco de dados u_i e o alvo t_i que supostamente se deseja representar e o custo de concatenação $C^c(u_{i-1}, u_i)$ de unidades sucessivas (HUNT, 1996).

A Figura B.3 ilustra um banco de dados de voz como uma rede de transição de estados. Os estados (caixas) representam todos os fonemas no banco de dados, organizados de acordo com a identidade fonética, e as linhas representam as transições, que são todas as sequências de concatenação possível.

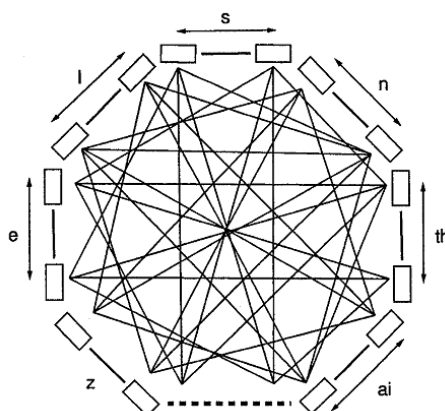



Figura B.3: banco de dados como uma rede de transição de estados. Fonte: (HUNT, 1996).

Dada a especificação-alvo, a sequência $t_1^n = (t_1, \dots, t_n)$, precisa-se selecionar o conjunto de unidades $u_1^n = (u_1, \dots, u_n)$ mais próxima ao alvo, minimizando o esforço de computacional de se aplicar técnicas de processamento de sinais para produzir as

características prosódicas exigidas bem como minimizar as distorções das formas de onda. O banco de dados contendo as unidades-candidatas pode ser visto como uma rede de transição de estados, com cada unidade representando um estado. O custo de permanência em um estado pode ser visto como o custo-alvo e o custo de transição de estados pode ser visto como o custo de concatenação. Como qualquer unidade pode ser potencialmente concatenada com qualquer outra, a rede é completamente conectada. O fonema-alvo é sempre sintetizado por uma unidade do banco de dados com a mesma identidade fonética (HUNT, 1996).

Cada alvo possui um *pitch*, duração e potência desejada. A tarefa é achar um caminho pela rede de transição de estados, a sequência no banco de dados de unidades, com custo mínimo (HUNT, 1996).

Cada fonema-alvo é um candidato no banco de dados e é caracterizado por um vetor de características multidimensional (HUNT, 1996).

O treinamento possui muitas similaridades como a síntese HMM (HUNT, 1996).

O treinamento para os custos-alvo e o custo de concatenação podem ser realizados ou com busca no espaço de pesos ou por regressão. Ambos os métodos usam vozes naturais e fornecem bons resultados (pesos) que quando treinados manualmente. Entretanto, entre essas duas técnicas, a regressão apresenta maior flexibilidade e menor custo computacional (HUNT, 1996).

O método tem sido aplicado para uma variedade de bancos de dados, incluindo para idiomas japonês e inglês e tanto para vozes masculinas como femininas (HUNT, 1996).

Cada custo-alvo é calculado como a soma ponderada das diferenças entre os vetores de características dos alvos e candidatos. O peso em geral varia de 20 a 30 (HUNT, 1996):

$$C^t(t_i, u_i) = \sum_{j=1}^{p} w_j^t C_j^t(t_i, u_i). \qquad (71)$$

O custo de concatenação é dado de forma semelhante (HUNT, 1996):

$$C^c(u_{i-1}, u_i) = \sum_{j=1}^{q} w_j^c C_j^c(u_{i-1}, u_i). \qquad (72)$$

Assim, o custo total para uma sequência de n unidades é a soma dos custos-alvo e de concatenação (HUNT, 1996):

$$C(t_1^n, u_1^n) = \sum_{i=1}^{n} C^t(t_i, u_i) + \sum_{i=2}^{n} C^c(u_{i-1}, u_i) + C^c(S, u_1) + C^c(u_n, S), \qquad (73)$$

em que S denota o silêncio, e $C^c(S, u_1)$ e $C^c(u_n, S)$ definem as concatenações de início e fim dadas pela concatenação da primeira e da última unidade com silêncio (HUNT, 1996).

A parte mais complexa é determinar os pesos das funções custo w_j^t e w_j^c.

O treinamento de regressão envolve a comparação exaustiva das unidades do banco de dados e regressão linear múltipla. A tarefa do treinamento é determinar os pesos que minimizem a diferença entre a forma de onda natural e a forma de onda gerada pelo sintetizador dada a especificação-alvo (HUNT, 1996).

A desvantagem do treinamento em busca no espaço de pesos é que o custo computacional cresce exponencialmente com o número de pesos a serem treinados e com o número de valores a serem usados com o peso, o que pode exigir mais de 150 horas de treino para um banco de dados de 40.000 unidades (aproximadamente 1 hora de diálogo) (HUNT, 1996).

A regressão determina os pesos dos custos de concatenação e alvo separadamente (HUNT, 1996).

Estudos têm mostrado que a combinação linear da distância cepstral e a diferença da potência no ponto de concatenação é um preditor razoável para a qualidade da concatenação (HUNT, 1996).

As vantagens da regressão são: capacidade de gerar de forma eficiente e separada pesos para diferentes classes de fonemas cujos contextos prosódicos são diferentes, e maior eficiência computacional. Entretanto, o treinamento pode levar de 1 a 10 horas, dependendo do banco de dados (HUNT, 1996).

B.4 Síntese baseada em modelos de Markov ocultos

A abordagem concatenativa se limita a recriar o que já foi pré-gravado. Uma alternativa é usar técnicas de síntese de parâmetros estatísticos para inferir especificações. Tais técnicas apresentam duas vantagens: exige-se menos memória para armazenar os parâmetros dos modelos do que os dados propriamente ditos. A outra vantagem reside no fato de garantir maior variabilidade: uma voz, por exemplo, pode ser convertida em outra (TABET, 2011).

Síntese de voz de parâmetros estatísticos tem crescido em popularidade nos últimos anos: a técnica consiste, basicamente, em gerar a média de um conjunto de segmentos de voz similares. Os resultados obtidos tanto em termos de naturalidade como grau de entendimento do que foi dito são bastante interessantes. O algoritmo se baseia na noção de custo-alvo, uma medida do quão adequado é um determinado candidato existente no banco de dados quando comparado com a unidade desejada. Juntamente com o custo do alvo, é definido também o custo de concatenação. O custo-alvo entre uma unidade u_i e uma unidade desejada t_i é dado por:

$$C^t(t_i, u_i) = \sum_{j=1}^{p} w^t{}_j \, C^t{}_j(t_i, u_i), \tag{74}$$

e o custo de concatenação é definido por:

$$C^c(u_{i-1}, u_i) = \sum_{k=1}^{q} w^c{}_k \, C^c{}_k(u_{i-1}, u_i), \tag{75}$$

em que $w^t{}_j$ e $w^c{}_k$ são pesos que podem ser definidos por uma combinação de treino e ajustes manuais. Unidades do mesmo tipo são agrupadas em uma árvore de decisão (BLACK, 2007).

Síntese de voz de parâmetros estatísticos oferece uma ampla gama de técnicas para melhorar a qualidade da voz. Seus modelos mais complexos - quando comparado com seleções de unidades sonoras padrão, permitem bons resultados para soluções gerais, sem necessidade de gravar todos os fonemas e contextos prosódicos (BLACK, 2007).

Dentre a síntese de parâmetros estatísticos, uma das técnicas mais usadas é a baseada em Modelos de Markov Ocultos (HMM - *Hidden Markov Models*). O modelo consiste de duas fases: a fase de treinamento e a fase de síntese. Durante a fase de treinamento deve-se decidir quais características os modelos devem treinar. Coeficientes mel-cepstrais (MFCC - *Mel Frequency cepstral coefficients*) de frequência e suas primeiras e segundas derivadas são as características mais usadas. O algoritmo de Baum-Welch é usado com os vetores de características para produzir modelos para cada fone. Um modelo consiste basicamente de três estados, representando o começo, o meio e o fim de um fone. A fase de síntese consiste de duas etapas: primeiramente, os vetores de características de uma dada sequência de fonemas devem ser estimados. Depois, um filtro é implementado para converter os vetores de características em sinais de áudio (TABET, 2011).

A síntese HMM é baseada em modelos de Markov ocultos. Neste sistema, o espectro em frequência (trato vocal), a frequência fundamental (fonte vocal) e duração (prosódia) da fala são modelados simultaneamente por HMM. As formas de onda são geradas pelo critério de máxima verossimilhança.

A síntese baseada em Modelos de Markov Ocultos tem mostrado ser bastante efetiva, gerando resultados satisfatórios. A técnica de síntese baseadas em HMMs tem recebido grande atenção também pela facilidade de aplicação e qualidade dos resultados dentre as técnicas mais recentes. Para aplicação de HMMs, podem ser usadas bases de voz com baixa qualidade ("caseiras") e poucas amostras e ainda assim, obter resultados satisfatórios (BLACK, 2007; COSTA e MONTE, 2012).

Na síntese por seleção de unidades, múltiplas instâncias de cada fone em diferentes contextos são armazenadas em um banco de dados. Construir tal banco de dados é uma tarefa custosa, além de resultar em um banco de dados grande (TABET, 2011).

O diagrama de blocos de um sistema baseado em HMM é mostrado na Figura D.4. O sistema é divido em duas partes: treinamento e síntese.

Na etapa de treinamento, um conjunto de HMMs (um por fonema) é treinado com parâmetros amostrais da voz e parâmetros contextuais prosódicos, a fim de gerar um modelo que relaciona regras contextuais prosódicas, com parâmetros amostrais da voz. Esta etapa inclui os seguintes sub-processos: geração de rótulos de contexto para cada frase da base; alinhamento forçado a nível de monofone para cada frase da base; reamostragem dos arquivos de áudio, se necessário, e conversão para o formato RAW.

Na etapa de síntese, módulos de NLP serão utilizados para gerar informações prosódicas de contexto, a fim de que as mesmas determinem a geração dos parâmetros amostrais da voz, que será a entrada para um filtro MLSA, responsável por gerar aproximações de voz baseado em parâmetros amostrais, criando assim, voz sintetizada (COSTA e MONTE, 2012).

A etapa de treinamento é semelhante àquela existente em sistemas de reconhecimento de voz. A principal diferença reside no espectro (coeficientes mel-cepstrais e sua dinâmica) e parâmetros de excitação ($logf_0$) que são extraídos a partir de um banco de dados e modelado por HMMs dependentes de contexto - contextos fonéticos, linguísticos e prosódicos são levados em consideração. A modelagem dos parâmetros envolvem distribuições de probabilidade multi-espaço ($logf_0$) e densidades de durações de estado para modelar estruturas temporais da fala. Assim, o sistema modela espectro, excitação e duração.

Os parâmetros alterados na etapa de treinamento são os seguintes: fator alfa, ordem de análise mel-cepstral e *frame shift*.

O fator alfa é relacionado à distorção da fala e é diretamente dependente da frequência de amostragem e também do locutor. Já a Ordem de análise mel-cepstral define a quantidade de padrões que serão analisador por quadro. Assim, maior a ordem, melhor o resultado da análise. Porém, é importante observar que para baixas taxas de amostragem pode ser até prejudicial uma análise muito grande, pois aumentando a análise, não estará aumentando q quantidade de informação nos padrões analisados. O *frame shift*, quando alterado na etapa de treino pode melhorar parte do resultado do modelo gerado. Na etapa de síntese, é o responsável por determinar a velocidade da fala (COSTA e MONTE, 2012).

Em geral, quanto maior for a frequência de amostragem usada para gravar as sentenças que compõem a base de treino, melhor é o resultado final. A explicação se deve pelo fato do modelo gerado pelo processo de aprendizagem conter mais informações.

A parte da síntese realiza a operação inversa do reconhecimento de voz: inicialmente o texto é marcado de acordo com rótulos dependentes de contexto. Em seguida, as durações dos estados do HMM são determinados de acordo com as funções de densidade de probabilidade das durações dos estados. Após esta etapa, o algoritmo de geração de parâmetros gera uma sequência de coeficientes mel-cepstrais e os valores de $log f_0$ que maximizam suas probabilidades de saída. Finalmente, a forma de onda da fala é sintetizada diretamente a partir dos coeficientes mel-cepstrais gerados e os valores de f_0 usando um filtro MLSA com pulso binário ou ruído de excitação (BLACK, 2007).

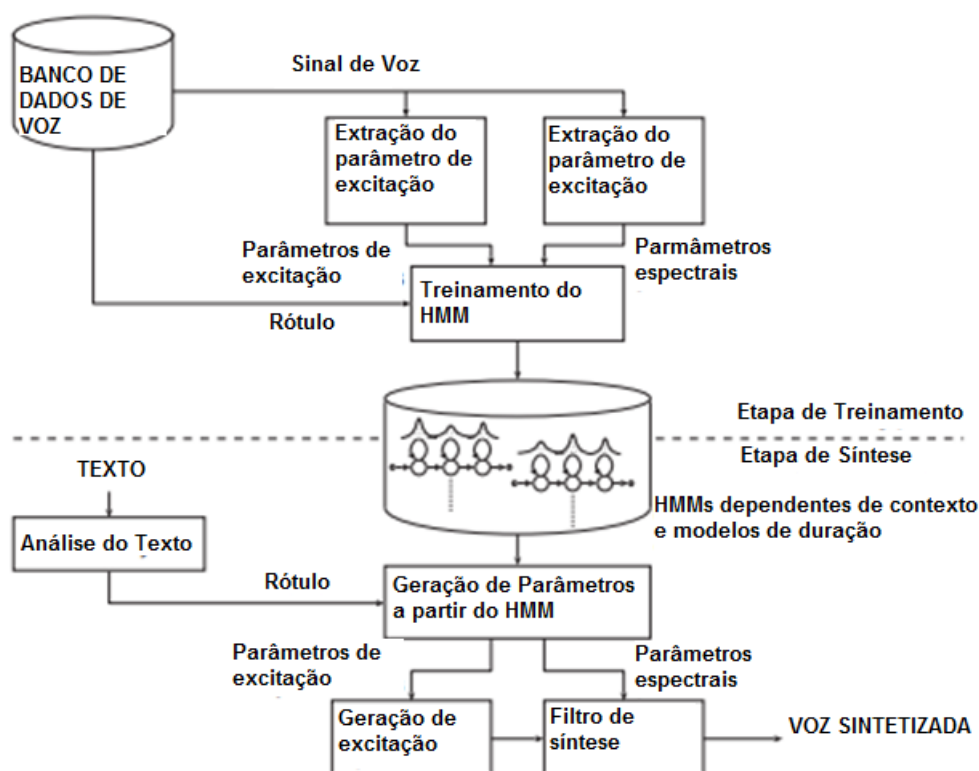



Figura B.4: visão geral de um sistema de síntese de voz baseado em HMM. Fonte: (BLACK et. Al. 2007 - Traduzido).

Em síntese baseado em HMM, distribuições para o espectro, f_0 e duração são agrupados independentemente, tendo portanto, para cada um deles, uma árvore de decisão diferente. As árvores de decisão para os dois últimos são equivalentes a árvores de regressão existentes nos sistemas de seleção de unidades sonoras (BLACK, 2007).

É possível também adotar abordagens híbridas. Algumas abordagens usam parâmetros espectrais, valores de f_0 e durações (ou parte deles) gerados a partir de HMM para calcular custos-alvos em seleção de unidades (BLACK, 2007).

B.4.1 Vantagens e desvantagens

As vantagens da síntese de voz baseado no HMM são: 1 - as características vocais são facilmente modificáveis, utilizando, por exemplo, interpolação; 2 - pode ser aplicado para diversas línguas, como japonês mandarim, coreano, inglês, alemão, português, sueco, esloveno, croata, árabe, etc., com poucas modificações; 3 - variações de estilos de fala ou emoções podem ser facilmente sintetizadas a partir de uma pequena quantidade de dados por meio de re-estimação da média dos modelos de voz existentes (BLACK, 2007).

A maior desvantagem dos algoritmos de síntese baseado em HMM são os três fatores que degradam a qualidade: *vocoder*, precisão na modelagem e suavização excessiva. Para o primeiro problema, alguns trabalhos propõem esquemas de excitação multi-banda ou STRAIGHT. Para o segundo problema, tem sido usados técnicas como HSMM, grafos estocásticos de Markov, critério de erro de geração mínima (MGE - *Minimum Generation Error*) e abordagem Bayesiana variacional. Em um sistema básico, o algoritmo de geração de parâmetros é usado para gerar parâmetros espectrais e de excitação a partir do HMM. Levando em conta restrições entre características estáticas e dinâmicas, o HMM pode gerar suavizações. Entretanto, os parâmetros espectrais e de excitação frequentemente são excessivamente suavizados. A fim de reduzir este efeito e melhorar a qualidade da fala, pós-filtragem, algoritmos de geração de parâmetros considerando variância global ou algoritmos de geração de parâmetros de voz condicionais podem ser utilizados (BLACK, 2007).

B.4.2 Estudos sobre variabilidade da voz em HMM

No HMM, modelos estatísticos do espectro e das características prosódicas são usadas para gerar uma voz sintética. Em sistemas HMM, vetores médios de modelos estatísticos são usados para gerar vozes sintéticas porém monótonas. No mundo real, é possível observar diferenças sensíveis na voz mesmo de um mesmo falante em diferentes instantes de tempo. Em termos técnicos, isto significa que a variância da distribuição é raramente examinada, gerando um problema de estabilidade excessiva. A variabilidade na voz humana, portanto, é um empecilho para os sistemas baseados em HMM convencional. Em (CHEN et. al., 2013) é proposta uma solução para lidar com a variabilidade da fala, que raramente recebe atenção nos estudos sobre o tema. O trabalho propõe um método capaz de gerar vozes humanas variantes no tempo e uma fala expressiva e diversa, diferente dos sistemas tradicionais que geram vozes, por vezes considerada por seus usuários como fria e monótona. Assim, Um *tradeoff* entre estabilidade e variabilidade deve ser considerado a fim de garantir uma melhor naturalidade.

Um diagrama de blocos de um sistema HMM tradicional combinado com a estratégia apresentada no trabalho é mostrado na Figura B.5. Durante a fase de treinamento, o espectro e os parâmetros de excitação são extraídos e modelados por HMMs dependentes de contexto. Durante a fase de síntese, um texto dado é convertido em uma sequência de rótulos dependentes de contexto por um analisador de texto.

A solução apresentada no trabalho citado apresentado considera que cada vetor da distribuição de estados é provavelmente um vetor de características da fala se o modelo de distribuição é preciso o suficiente e que a probabilidade de saída de um vetor com a distância mínima do vetor médio é maior.

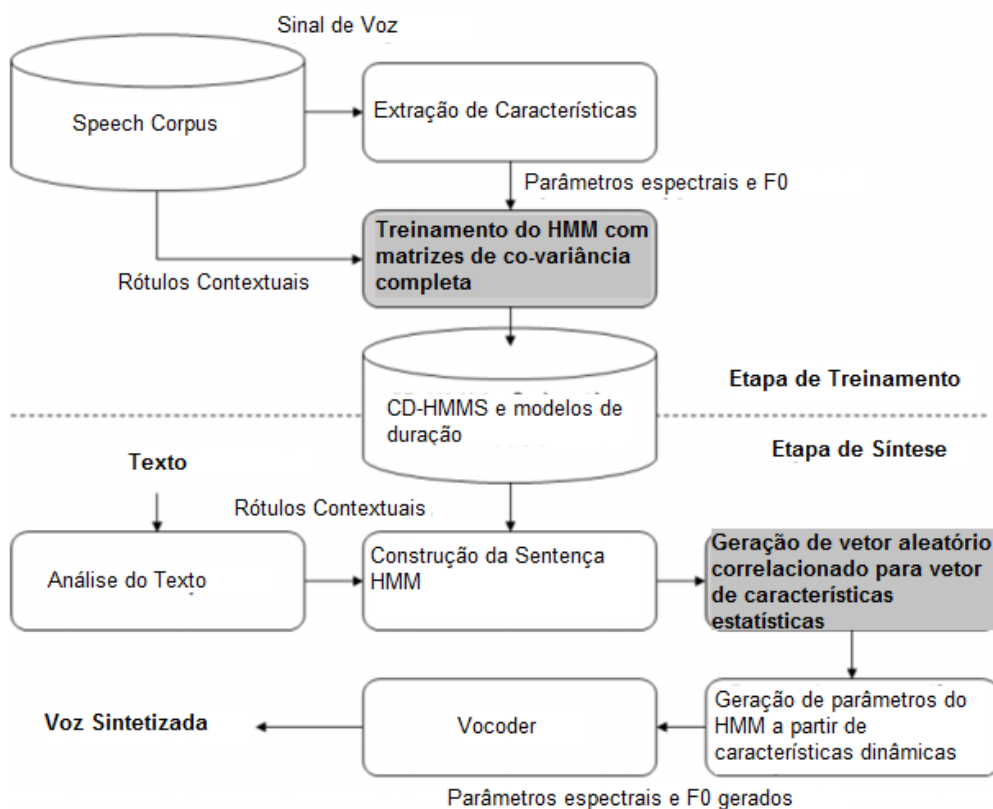



Figura B.5: solução apresentada em (CHEN et. al., 2013) para garantir variabilidade na voz. Fonte: (CHEN et. al., 2013 - Traduzido).

Infelizmente, o modelo de distribuição tradicional da síntese de voz não é preciso o suficiente porque uma distribuição gaussiana multivariada com matriz de covariância diagonal é geralmente utilizada ignorando a correlação das dimensões por razões de baixo custo computacional e armazenamento de dados. Entretanto, ignorar correlações dimensionais resulta em um modelo de distribuição impreciso. Assim, a fim de aprimorar o modelo de distribuição, a matriz de covariância completa deve ser considerada. Entretanto, uma matriz de covariância completa é difícil de estimar devido ao grande número de parâmetros livres. Para resolver este problema, usa-se então a Transformação Linear de Máxima Verossimilhança (MLLT - *Maximum Likelihood Linear Transformation*) a fim de estimar as matrizes de covariância completas. MLLT visa minimizar a perda na verossimilhança entre a função de distribuição de

probabilidade gaussiana covariância completa e as diagonais. Na fase de treinamento, MLLT estima a matriz de covariância completa a fim de garantir que cada característica é independente. Na fase de síntese, geram-se características para os parâmetros de síntese de voz em cada estado, usando vetor médio e a matriz de correlação das funções de distribuição de probabilidade dos estados (CHEN et. al., 2013).

B.4.3 Modelagem de matrizes de precisão por Transformação Linear de Máxima Verossimilhança

Correlações entre todas as características não podem ser obtidas através de matriz de covariância diagonal. Entretanto, a fim de realizar uma aproximação adequada os modelos de covariância completa, MLLT é introduzido no estágio de treinamento do HMM. Cada matriz de precisão da distribuição Gaussiana para as funções de distribuição de probabilidades das saídas de estado é igual ao inverso da matriz de covariância da j-ésima distribuição gaussiana (CHEN et. al., 2013):

$$P_j = A^T \Lambda_j A, \tag{76}$$

em que P_j é a distribuição de probabilidades dos estados de saída, A representa a matriz de transformação global, Λ representa a matriz diagonal de distribuição específica cujos elementos da diagonal principal são os inversos das variâncias no espaço transformado ($\Lambda_j^{ii} = 1/\sigma_{ji}^2$).

B.4.4 Geração de parâmetros de fala com geração de vetores aleatórios correlacionados

No estágio de treinamento, a matriz de covariância completa é estimado via MLLT. No estágio de síntese, após a sentença HMM ser construída, a sequência de estados é obtida:

$$\boldsymbol{q} = \{q_1, q_2, \dots, q_T\}, \tag{77}$$

em que q_t indica o t-ésimo estado da sequência q. Cada estado q_t consiste de um vetor médio M-dimensional da característica estática $\boldsymbol{c} = [\boldsymbol{c}_t(1), \boldsymbol{c}_t(2), \dots, \boldsymbol{c}_t(M)]^T$, A matriz de transformação global A *MxM,* Λ_j a matriz de covariância diagonal *MxM*, e os vetores médios M-dimensionais das características dinâmicas $\Delta\boldsymbol{c}_t$:

$$\Delta\boldsymbol{c}_t = 0.5(\boldsymbol{c}_{t+1} - \boldsymbol{c}_{t-1}), \tag{78}$$

Como a matriz de covariância do vetor de parâmetros estáticos é uma matriz positiva semi-definida, então $\sum$ pode ser expresso como $\sum = U^T D U = (\sqrt{D}U)^T (\sqrt{D}U)$. A decomposição de Cholesky de $\sum$é $\sum = C^T C$. (CHEN et. al., 2013) mostra então que $\sum_j = (\sqrt{\Lambda_j}A)^{-1} (\sqrt{\Lambda_j}A)^T$, o que nos dá então $C = \left((\sqrt{\Lambda_j}A)^{-1}\right)^T$. Sendo o vetor

aleatório $\boldsymbol{x} = \boldsymbol{C}^T\boldsymbol{r} + \boldsymbol{\mu}$ reescrito como $\boldsymbol{c}_t^* = (\sqrt{\boldsymbol{\Lambda}_j}\boldsymbol{A})^{-1}\boldsymbol{r} + \boldsymbol{c}_t$ em que $\boldsymbol{r}$ é o vetor aleatório M-dimensional e $\boldsymbol{\mu}$ é o vetor médio da distribuição Gaussiana.

B.5 Síntese baseada em grafos de Markov

A vantagem de usar Grafos Estocásticos de Markov (SMG - *Stochastic Markov Graphs*) ao invés de HMMs em síntese de voz paramétrica reside na capacidade melhorada dos SMGs modelarem trajetórias em um espaço de características. Sintetizadores baseados em SMGs requerem menos espaço de armazenamento do que a síntese concatenativa. Embora a síntese baseada em SMGs não apresente a mesma qualidade que a síntese baseada em concatenação atualmente, espera-se que a qualidade de ambos deva se equiparar em um futuro próximo (EICHNER, 2001).

Seja $\gamma(U, \Psi_{UU})$ um grafo dirigido com estados (vértices) $U = \{u_1, u_2 \dots u_N\}$ e a relação de incidência $\Psi_{UU}: UxU \rightarrow \{\emptyset, 1\}$. Denotamos uma aresta entre dois estados u_i e u_k como $(u_i \rightarrow u_k)$. Uma aresta é definida unicamente pela relação de incidência. A probabilidade de transição do arco $(u_i \rightarrow u_k)$, estimado no processo de treinamento, é escrito como $P(\Psi_{u_i,u_k})$. A sequência de sucessivas arestas em um grafo é chamado de caminho q. O i-ésimo estado do caminho q é denotado por $q(i)$ (EICHNER, 2001).

Começa-se o treinamento com uma estrutura HMM convencional. No estágio de inicialização, cada estado é assinalado com uma distribuição Gaussiana. Após o treinamento, cada estado do modelo é dividido em dois. Então, as arestas e os caminhos improváveis são removidos do SMG. Tal processo é realizado em dois estágios: no primeiro, todas as arestas com probabilidade de transição inferior a um dado limiar *p* são removidos do grafo. Tais procedimentos são repetidos até que se atinja um número máximo total de estados ou o número de estados descartados no último estágio da iteração seja maior que $0{,}3 \cdot 2^I$, em que I é o número de divisões de estados desde o início (EICHNER, 2001).

A síntese é realizada por meio das seguintes etapas: selecionar uma sequência de estados usando SMG de acordo com a duração do fonema-alvo (comprimento da sequência solicitada) e a modelagem da duração de cada estado no caminho; montagem da sequência de vetores característicos para o caminho escolhido por meio de extração das médias das Gaussianas correspondentes; geração de sinal de voz usando filtro MLSA (EICHNER, 2001).

No primeiro estágio, transforma-se os SMGs treinados gama em uma apresentação alternativa gama' por meio da transformação TE (*Tree Expansion*) (EICHNER, 2001):

$$\gamma'(U', \Psi'_{U'U'}) = TE\big(\gamma(U, \Psi_{UU})\big). \tag{79}$$

Entretanto, os SMGs contém laços, sendo então necessário modificar a expansão em árvore, utilizando o algoritmo explicado em (EICHNER, 2001).

A Figura B.6 mostra o fluxograma do algoritmo proposto em (EICHNER, 2001).

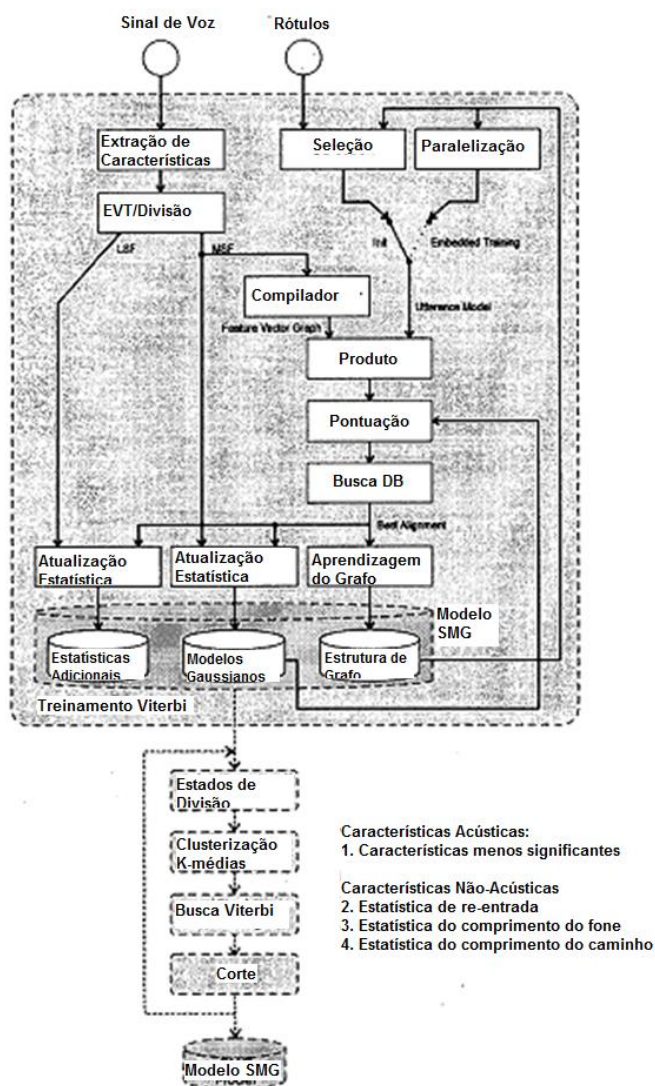



Figura B.6: Funcionamento da síntese SMG. Fonte: (EICHNER, 2001 - Traduzido).

B.6 Síntese HNM

Modificações prosódicas da fala são necessárias para se produzir sistemas com alta qualidade de síntese. Modelos HNM são modelos paramétricos e apresentam facilidade para modificar características prosódicas como entonação, estresse ou ritmo (TABET, 2011).

HNM assume que o sinal de fala é composto por uma parte harmônica e uma parte de ruído. A parte harmônica corresponde aos componentes quasi-periódicos da fala e o ruído corresponde aos componentes não periódicos. Tais componentes são separadas no domínio da frequência por um parâmetro chamado frequência máxima vozeada F_m. A largura de banda até F_m é representada por senoidais harmônicas e a largura de banda a partir de F_m é representada por componentes de ruído modulados. Sons não vozeados são representados apenas pela parte do ruído. O sinal de voz é então obtido a partir da soma das harmônicas com ruído.

A parte harmônica contém apenas múltiplos da frequência fundamental. A parte do ruído pode ser modelada a partir de um envelope usando filtro AR, no qual a síntese é realizada por meio da filtragem de ruído branco pelo filtro AR. A fase dos ruídos é ajustada aleatoriamente devido ao fato do ruído ser um sinal estocástico (TABET, 2011).

A parte periódica (ou quasi-periódica) é considerada harmônica. Nas primeiras implementações do HNM, a parte harmônica designava a soma de componentes senoidais harmonicamente relacionadas com amplitudes constantes dentro de cada quadro. A fase é modelada por um polinômio de primeira ordem - por exemplo, pressupõe-se que seja linear. Versões posteriores consideravam a parte harmônica também como a soma de componentes senoidais harmonicamente relacionadas porém com amplitudes complexas variando linearmente. Outras implementações usavam um polinômio de grau p com coeficientes reais para as amplitudes harmônicas e assumiam que as fases eram lineares.

Dada a parte harmônica, a parte aperiódica é obtida subtraindo a parte harmônica do sinal de voz original. A parte aperiódica , ou sinal residual, é considerada em todos os sinais não descritos por componentes harmônicas. Isto inclui ruídos fricativos, flutuações entre períodos produzidos pela turbulência do fluxo de ar glotal, etc. (TABET, 2011).

A qualidade do resultado gerado por HMM não é tão alta quanto na síntese por seleção de unidades. A precisão da modelagem pode ser melhorada usando técnicas como Modelos semi-Markov Ocultos e grafos estocásticos de Markov, por exemplo ou então integrar sistemas HTS (*Hidden Markov Model-based Speech Synthesis System*) ao HNM. Tal integração reduz o tempo de desenvolvimento e custo em comparação com técnicas do estado-da-arte baseado em seleção automática e síntese concatenativa, produzindo resultados melhores quando comparado ao HTS. Tal qualidade é alcançada

substituindo a abordagem da modelagem do filtro da fonte usada no HTS com pelo modelo HNM, conhecido por ser capaz de produzir respostas mais naturais sob várias modificações da prosódia (TABET, 2011).

B.7 Síntese LPC

A síntese de formantes provê uma arquitetura flexível, mas requer a especificação de diversos valores para modelar a coarticulação, exigindo especialistas capazes de manipular todos os parâmetros de síntese. A síntese LPC apresenta o uso de uma estrutura mais simples: todos os detalhes da voz modelados, exceto intensidade e periodicidade, são inclusos nos coeficientes dos filtros LPC. Filtros digitais são evitados devido à sensibilidade ao ruído de quantização e o risco de instabilidade (SHAUGHNESSY, 2003).

Atualmente, a síntese baseada em codificação preditiva (LPC - *Linear Predictive Coding*) tem chamado atenção por sua baixa taxa de dados, baixa complexidade e baixo custo, entretanto, devido os parâmetros extraídos a partir de um modelo original é simples demais para produzir resultados de alta qualidade.

B.7.1 Aplicação do algoritmo AMR-WB para síntese LPC

A tecnologia de codificação de voz AMR-WB (*Adaptative Multi-Rate Wideband*), usando Predição Linear Excitada de Código Algébrico (ACELP - *Algebric Code Excited Linear Prediction*) e combinação de técnicas ara calcular o atraso do *pitch* no estágio de extração de parâmetros tem se apresentado como uma alternativa viável e de alta qualidade para melhoria do LPC. A síntese realiza uma etapa de pré-processamento que inclui decimação a fim de reduzir a quantidade de dados a serem processados, um filtro passa alta e uma pré-ênfase. Em seguida, o sistema passa por uma etapa de quantização, que é feito encontrando-se o índice k que minimiza $E = \sum[\gamma_i - \hat{\gamma}_i^k]^2$ em que γ_i é o sub-vetor de erro residual e $\hat{\gamma}_i$ é o vetor quantizado para o índice k. Por fim, é realizada a extração do *pitch* e a construção da excitação.

Em termos de complexidade computacional, o algoritmo de extração de parâmetros AMR-WB apresenta alta complexidade, entretanto, tal complexidade é compensada pela alta qualidade do resultado, quase não apresentando diferenças, tanto no domínio do tempo como da frequência, com relação a um sinal de voz amostrado. (SHU, et. al. 2011).

B.8 Outras abordagens

A síntese de formantes e a síntese articulatória são menos usados atualmente, sendo utilizados mais atualmente técnicas como a síntese de seleção de unidades combinado com HNM, representando o sinal como a soma de harmônicos com ruído, uma vez que a decomposição do sinal de voz nessas duas partes permite modificações mais naturais da fala, além de suavizar as descontinuidades das unidades acústicas. A maior limitação dessa combinação reside no elevado custo computacional (TABET, 2011).

Esta combinação tem produzido resultados satisfatório e consumindo pouca memória para armazenar parâmetros quando combinada com HMM, permitindo, além disso, maior variabilidade (TABET, 2011).

Já outros trabalhos tem procurado conciliar HNM e HMM a fim de reduzir custos e tempo de desenvolvimento (TABET, 2011).

B.8.1 A abordagem proposta em (BRAUNSCHWEILER, 2010)

Nos modelos clássicos, mesmo com grandes bancos de dados, descontinuidades e prosódias pouco naturais causadas por escolhas inadequadas entre o alvo e a unidade selecionada são inevitáveis. Por outro lado, os métodos que modificam a frequência fundamental geram prosódia precisa para sotaques e entonações, mas podem produzir vozes pouco naturais, robóticas, degradando a qualidade devido a modificações prosódias (BRAUNSCHWEILER, 2010).

O algoritmo descrito em (BRAUNSCHWEILER, 2010) visa reduzir a degradação por modificações prosódicas e descontinuidades por meio de um método de síntese que combina concatenação de formas de onda naturais e uma técnica própria de seleção plural e fusão de unidades, modificando a frequência fundamental a e duração dos fones, capaz de regenerar a prosódia a partir das unidades selecionadas e usando múltiplas unidades em segmentos não adjacentes, reduzindo as descontinuidades, apresentado resultados superiores aos métodos convencionais. A entrada do sistema é uma sequência de fonemas e a prosódia.

O método consiste em selecionar múltiplas unidades de voz para cada segmento de semi-fonemas, e então gerar formas de onda que representam as múltiplas unidades realizando uma média das formas de onda em um ciclo de *pitch*. Tal solução permite a suavização da concatenação de segmentos não adjacentes mantendo a qualidade. Então a prosódia é regenerada a partir de uma única (ou múltiplas) unidade(s) a fim de reter as expressões prosódicas da fala original. Finalmente, a prosódia das unidades é

modificada de acordo com a prosódia regenerada e então as unidades geradas são concatenadas a fim de produzir uma fala (BRAUNSCHWEILER, 2010).

A Figura B.7 mostra um diagrama de blocos para a solução proposta. O banco de dados de unidades sonoras contém informações sobre forma de onda dos segmentos, marcadores de *pitch*, atributos prosódicos, atributos de contextos fonéticos e atributos de contextos gramaticais. São usados semi-fonemas como as menores unidades sonoras. A sequência de fonemas, a prosódia gerada no módulo de geração de prosódia juntamente com informações de atributos para a seleção de unidades são usadas como entradas (BRAUNSCHWEILER, 2010).

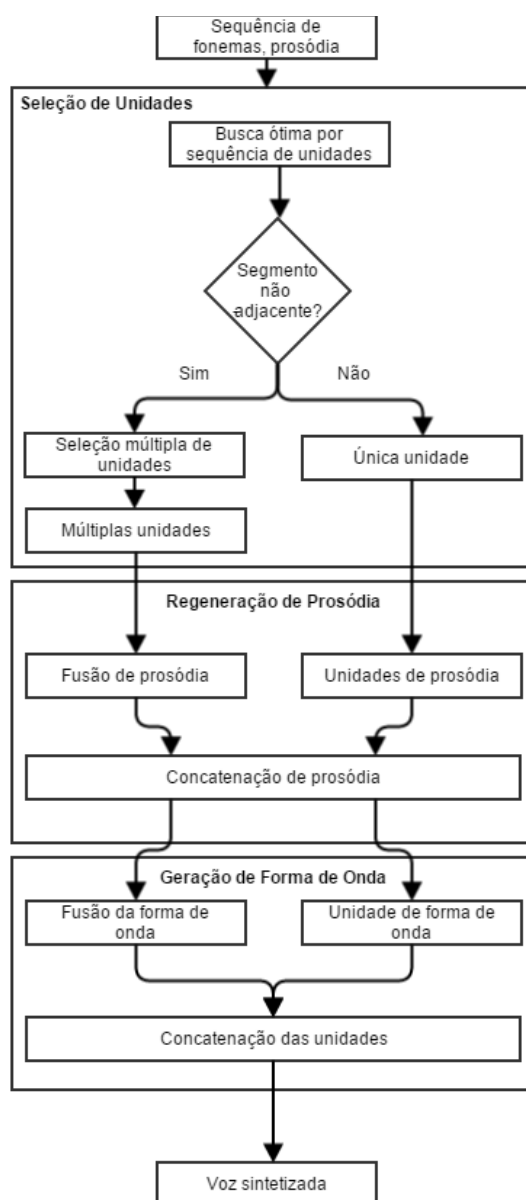



Figura B.7: solução proposta por (BRAUNSCHWEILER, 2010). Fonte: (BRAUNSCHWEILER, 2010 - Traduzido).

Tal técnica faz uso dos custos alvo e dos custos de concatenação (BRAUNSCHWEILER, 2010).

Na etapa de seleção de unidades, segmentos de cada semi-fonema são selecionados. A unidade ótima é selecionada usando uma função custo que consiste dos custos-alvo e de concatenação.

Neste caso, são definidos como a soma ponderada do custo da frequência fundamental, custo da duração do alvo, custo do contexto fonético e custo do contexto gramatical. O custo de concatenação é definido como o custo de concatenação da frequência fundamental, o custo de concatenação do espectro, o custo de concatenação de potência e o custo de adjacência (0 quando as unidades são adjacentes e 1 caso contrário). No referido trabalho, os pesos foram ajustado manualmente. O custo de contexto gramatical são calculadas as distâncias das sílabas no começo e fim da sentença, grupo respiratório e palavras, justamente com a distância das sílabas acentuadas em uma palavra (BRAUNSCHWEILER, 2010).

Na etapa de regeneração da prosódia, a duração dos fonemas e o contorno da frequência fundamental são regenerados usando as unidades selecionadas. A média do contorno da frequência fundamental é suavizada e concatenada e é realizada por meio de adição de um valor de deslocamento (*off-set*), interpolação linear e *spline*. O valor de deslocamento desloca o contorno da frequência fundamental para reduzir a diferença na fronteira.

Na etapa de geração de forma de onda, formas de onda que foram geradas a partir das múltiplas unidades selecionadas são usadas nos segmentos não adjacentes. E finalmente então, a forma de onda é sintetizada usando tais formas de onda (BRAUNSCHWEILER, 2010).

Na etapa de regeneração de prosódia, durações e contornos da frequência fundamental são geradas a partir das unidades. A duração para segmentos não adjacentes são geradas a partir da média das durações das unidades selecionas e calculadas pela expressão:

$$d_{syn}^{i} = \frac{1}{N}\sum_{n=1}^{N} d_{n}^{i}, \tag{80}$$

em que d_{syn}^{i} e d_{n}^{i} representam a duração gerada e a duração da n-ésima unidade selecionada para o i-ésimo segmento, respectivamente. Para os contornos da frequência fundamental, estes são gerados mapeando frames da frequência fundamental de cada

unidade e realizando a média deste mapeamento para cada frame. Ou seja, o contorno da frequência fundamental é dado por:

$$f_{0_{syn}}^{i}(t) = \frac{1}{N}\sum_{n=1}^{N} f_{0_n}^{i}(t\frac{d_n^i}{d_{syn}^i}), \quad (81)$$

em que $f_{0_{syn}}^{i}(t)$ e $f_{0_n}^{i}(t)$ representam o f_0 gerado e o f_0 da n-ésima unidade escolhidas para o i-ésimo segmento no tempo t, respectivamente.

O $\hat{f}_{0_{syn}}^{i}(t)$ para o i-ésimo segmento é dado por:

$$\hat{f}_{0_{syn}}^{i}(t) = f_{0_{syn}}^{i}(t) + offset^{i}, \quad (82)$$

O valor do f_0 médio do ponto final do semi-fonema esquerdo $f_{0_{syn}}^{i}(T)$ e o início do semi-fonema direito $f_{0_{syn}}^{i+1}(0)$ são calculados. O valor do *offset* é determinado como se segue:

$$offset^{i} = 0{,}5\left(f_{0_{syn}}^{i}(T) + f_{0_{syn}}^{i+1}(0)\right) - f_{0_{syn}}^{i}(T), \quad (83)$$

$$offset^{i} = 0{,}5\left(f_{0_{syn}}^{i-1}(T) + f_{0_{syn}}^{i}(0)\right) - f_{0_{syn}}^{i}(0), \quad (84)$$

para os fonemas esquerdo e direito, respectivamente.

A suavização dos contornos é feita minimizando a função erro definida por:

$$E = p\|\boldsymbol{y} - \boldsymbol{s}\|^2 + (1-p)\boldsymbol{s}'^{\boldsymbol{D}'_k\boldsymbol{D}_k}\boldsymbol{s}, \quad (85)$$

em que $\boldsymbol{y}$, $\boldsymbol{s}$,p, $\boldsymbol{D}_k$ representam o contorno de f_0 suavizado, o contorno da entrada f_0, parâmetro de suavização e uma matriz que fornece uma função diferencial de k-ésima ordem (BRAUNSCHWEILER, 2010).

B.8.2 A abordagem proposta em (PHUNG et. al.)

Seleção de unidades requer uma grande quantidade de dados para concatenação. O trabalho de (PHUN *et. al.*) propõe decomposição temporal para modelar efeitos contextuais inter e intra-sílabas, adequando a modificação e seleção das unidades de acordo com o contexto aplicado a línguas monossilábicas, mais especificamente, o idioma vietnamita. O algoritmo de síntese é mostrado na Figura D.8.

O método apresenta uma proposta para estimar as posições e a duração do núcleo e os intervalos de transição dentro de cada fonema. Em seguida, é aplicado um modelo para coarticulação acústica que representa os efeitos contextuais inter e intra-sílabas. Após esta etapa, usando o referido modelo, um método de modificação de unidades para adequar ao contexto é aplicado em conjunto com método de seleção de unidades

sensível a contexto. Por fim, a solução é integrada a um sistema CSS para línguas monossilábicas (PHUNG et. al.).

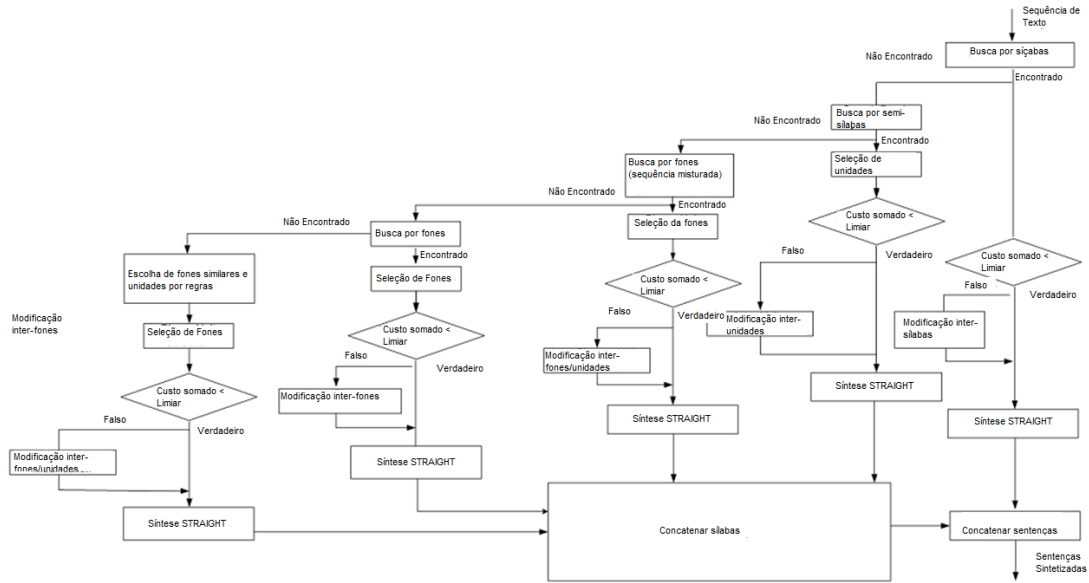



Figura B.8: algoritmo de síntese proposto em (PHUNG et. al. - Traduzido).

Para se determinar a posição e a duração dos núcleos e dos intervalos de transição dentro de uma sílaba, usou-se a medida de transição espectral (STM - *Spectral Transition Measure*). Para interpolar a fala e modificar a junção dos intervalos de transição usou-se TD de segunda ordem restrito modificado (MRTD - *Modified Restricted Second Order TD*). O STM no tempo t - o local do frame no domínio do tempo, é dado por (PHUNG et. al.):

$$STM(t) = \frac{1}{p}\sum_{i=1}^{p} a_i^2, \tag{86}$$

em que:

$$a_i = \frac{\sum_{n=-n_0}^{n_o} C_i(n)n}{\sum_{n=-n_0}^{n_o} n^2}, \tag{87}$$

$C_i(n)$ é o coeficiente espectral de i-ésima ordem ($1 \leq i \leq p$) no n-ésimo frame dentro da janela centrada em t, com $-n_o \leq n \leq n_o$. O coeficiente de regressão a_i, corresponde à variação linear do padrão do envelope espectral em uma unidade de tempo. Assim, $STM(t)$, que é a o valor quadrático médio de a_i corresponde à varoação do envelope espectral suavizado. Como o próprio nome indica, $STM(t)$ apresenta a medida da transição espectral em uma fala contínua (PHUNG et. al.).

APÊNDICE C: APIs PARA DESENVOLVIMENTO DE *SOFTWARES* BASEADOS EM VOZ

C.1 GNOME

GNOME é um sistema de Desktop avançado para usuários voltado para alguns sistemas derivados do Unix, como GNU/Linux e Solaris. Trata-se de um projeto *open source* e que segue o modelo de *software* livre. É um ambiente fácil de usar e altamente personalizável.

O projeto GNOME foi desenvolvido pensando no usuário, incluindo portadores de necessidades especiais, com problemas de visão, surdez ou motores. O GNOME oferece uma plataforma robusta e confiável para desenvolver aplicações acessíveis e interfaces para tecnologias assistivas e inclui leitor de tela, lente de aumento, etc.

O GNOME foi projetado desde seu início levando em considerações questões de acessibilidade e fornece um framework robusto que torna o desenvolvimento de aplicações acessíveis muito mais fácil. Além disso, provê uma interface padrão para integrar tecnologias assistivas, como leitores de tela e lentes de aumento virtuais. Tal interface é chamada de *Assistive Technology Service Provider Interface* (AT-SPI), que fornece uma ponte entre o AT-SPI e as aplicações baseadas em Java que fazem uso de componentes de interface com o usuário Swing.

A Figura C.1 mostra a arquitetura geral do GNOME no que diz respeito a algumas soluções voltadas para desenvolvimento de aplicativos acessíveis.

ARQUITETURA DE ACESSIBILIDADE DO DESKTOP GNOME

Camada de Ferramentas
Accessibility Helper
Bobby HTML tool
ATK test tool
Camada de Aplicação
StarOffice™ & OpenOffice.org
UNO access API
glue
Java™ Apps
JA-API
Netscape™
gecko engine
nslAccessible
nstAcc. extensions
ATK
GNOME desktop (a GTK+ App)
GTK+ Apps
GTK+
ATK
GAIL
Java bridge
ATK bridge
AT SPI (ORB)
API calls
Modern Window Mgr.
Gnopernicus Screen Reader/Magnifier
GOK On-Screen Keyboard
Magnification in X
Braille drivers
Switch drivers
Braille drivers
Switch drivers

O *Java Bridge* permite que aplicações baseadas em tecnologia Java usem as tecnologias assistivas fornecidas pelo GNOME Desktop.

O AT-SPI é a interface-padrão para integração de tecnologias asssistivas ao sistema.

O *ATK bridge* automaticamente conecta aplicações gnome gonstruídas com GTK+ com tecnologias assistivas instaladas no sistema.

Diversos tipos diferentes de tecnologias assistivas são suportadas pela arquiteturade acessibilidade GNMOE. Eles se comunicam unicamente pelo AT-SPI

Figura C.1: arquitetura do GNOME 2.0. Fonte (SUN MICROSYSTEMS, 2003 - Traduzido).

C.2 IBM ViaVoice TTS SDK

O IBM ViaVoice TTS SDK fornece aos programadores as ferramentas necessárias para o desenvolvimento de aplicações que incorporam a tecnologia de voz, incluindo um conjunto da APIs e utilitários que permitem ao desenvolvedor grande capacidade de padronização e gerenciamento do processo de síntese de voz acessado por uma aplicação. Este SDK permite aos desenvolvedores a escolha entre duas APIs distintas: *Eloquence Command Interface* (ECI) e *Microsoft Speech Application Programming Interface* (SAPI). O IBM ViaVoiceTTS SDK, juntamente com o IBM ViaVoice TTS Runtime, fornecem todos os *softwares* e arquivos de suporte para as duas APIs. ECI é uma API proprietária e independente de plataforma, que permite acesso direto a toda a funcionalidade do IBM ViaVoice TTS. Como características desta API destacam-se o seu suporte a diversos sistemas operacionais, padronização da saída de voz através de chamadas de funções e de anotações textuais, além de não utilizar o Registro do Windows para localização de componentes, evitando modificações acidentais de instalações por outras aplicações. SAPI é a API da Microsoft sendo suportada somente em sistemas Windows. Esta API fornece compatibilidade com padrões como ActiveX,

COM, DCOM, MSAgent, e também permite padronização da saída de voz por meio de chamadas de funções e marcações de texto SAPI.

O IBM ViaVoice TTS SDK é capaz de interpretar abreviaturas, acrônimos e números com alta qualidade e entonação bastante natural. Além disso, é possível inserir marcações no texto a fim de ajustar melhor a entonação e escolher o modo de interpretar textos e números, permitindo controlar atributos como ênfase em certas palavras e velocidade, personalizando a saída. É possível também utilizar uma ortografia fonética a fim de especificar a pronúncia de uma palavra.

O IBM ViaVoice TTS SDK fornece pelo menos cinco vozes predefinidas para cada idioma e cada uma tem uma marcação de voz correspondente que pode ser inserida no texto. Vozes individuais derivam sua exclusividade de diversos fatores físicos. Ademais, a voz de um indivíduo pode assumir formas diferentes de acordo com fatores como estado de espírito e circunstâncias. Estes atributos, tais como, trato vocal, linha de base de tom, dimensões do crânio, rouquidão, respiração, flutuação do tom, velocidade e volume podem ser modificados com um conjunto de marcações de caraterísticas de voz.

O IBM ViaVoice TTS permite que se especifique pronúncias explícitas para palavras abreviaturas e acrônimos, por meio de dicionários voltados para casos específicos: Dicionário de Palavras Especiais, Dicionário de Abreviaturas e Dicionário de Radicais.

C.3 Java Accessibility API

A maioria das tecnologias de voz estão implementadas em C e ++ e são voltadas para plataformas específicas, como a *Apple Speech Manager* e *Microsoft's Speech API* (SAPI) ou outras APIs proprietárias. (SUN MICROSYSTEMS, 1998).

Sintetizadores e reconhecedores de voz escritos em Java podem beneficiar da portabilidade da plataforma Java e das suas melhorias contínuas principalmente com relação à velocidade de execução da *Java Virtual Machine* (JVM). (SUN MICROSYSTEMS 1998).

A API *Java Accessibility* contém classes e interfaces que, quando aplicadas, garantem ao *software* tornar-se acessível às tecnologias assistivas (SANTOS, 2010).

A tecnologia Java conta com recursos que fornecem suporte à acessibilidade, tendo sido introduzida na linguagem a partir de março de 1996 e está apoiada em quatro áreas:

API *Java Acessibility*, *Java Acessibility Utilities*, *Java Accessibility Bridge* e *Pluggable Look and Feel* do *Java Foundation Classes* (SANTOS, 2010).

A API *Java Accessibility* define o contrato entre os componentes de interface do usuário e uma tecnologia assistiva para o acesso a esse aplicativo Java. Se um aplicativo Java suporta por completo a API *Java Accessibility*, então o mesmo é compatível com as tecnologias assistivas, como leitores e ampliadores de tela, etc.

Além da API de acessibilidade, existem também o *Java Accessibility Utilities*, fornecendo suporte necessário para as tecnologias assistivas na localização dos objetos que implementam a API *Java Accessibility* (SANTOS, 2010).

No que diz respeito a *Java Accessibility Bridge*, esta funciona como uma ponte entre a JVM e o ambiente nativo. Para que as tecnologias assistivas disponíveis nos sistemas operacionais possam fornecer acesso aos aplicativos Java, eles precisam de alguma forma para se comunicar com o suporte de acessibilidade Java. O *Java Accessibility Bridge* suporta essa comunicação (SANTOS, 2010).

A Figura C.2 mostra como é feita a comunicação entre O *Java Accessibility Bridge*, a aplicação Java, as classes utilitárias de acessibilidade e outras tecnologias assistivas.

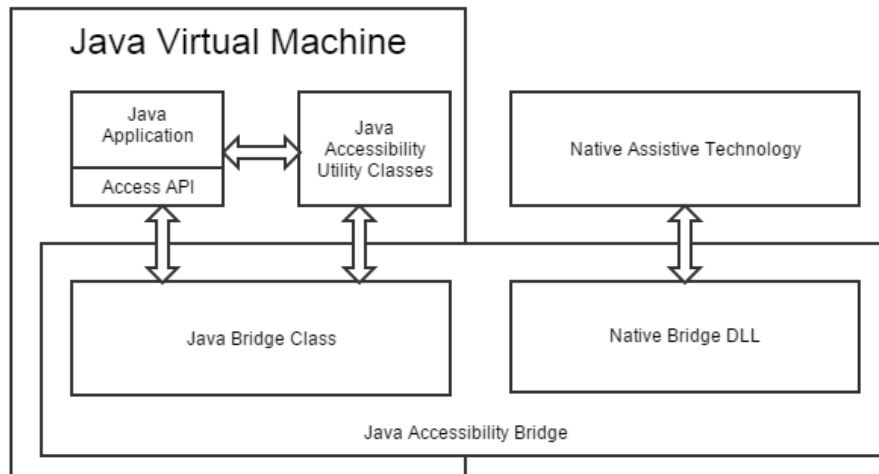



Figura C.2: diagrama de Funcionamento do Java Accessibility Brige. Fonte: (SANTOS, 2010 - Adaptado).

Para que uma aplicação possa ser considerada acessível, existe um conjunto de critérios que precisam ser atendidos. Um *checklist* de acessibilidade foi criado para as aplicações desenvolvidas com a tecnologia Java que se fundiu ao *checklist* de acessibilidade de produtos da IBM e é mostrado no "Anexo A" (SANTOS, 2010).

Especificamente, a API *Java Accessibility* define um contrato entre os componentes de interface usados em uma aplicação e a tecnologia assistiva que permite o acesso a essa aplicação Java. Se uma aplicação suporta totalmente a API *Java Accessibility*, então a mesma é compatível com leitores de tela, ampliadores de tela, e outros tipos de tecnologias assistivas (SANTOS, 2010).

É importante ressaltar que, para proporcionar a acessibilidade aos aplicativos escritos na linguagem de programação Java, uma tecnologia assistiva exigem mais do que a API de acessibilidade Java. Também requer um mecanismo para localizar os objetos que implementam essa API, bem como suporte para carregá-la na Máquina Virtual Java, rastreamento de eventos, e assim por diante. Portanto, a API *Java Accessibility* trabalha em conjunto com *Java Accessibility Utilities* para essa assistência (SANTOS, 2010).

Somente o uso da API *Java Accessibility* não é suficiente para prover a acessibilidade, sendo necessário utilizar o pacote de utilitários para fornecer apoio à API (SANTOS, 2010).

C.4 Java Speech API

A *Java Speech API* (JSAPI) foi desenvolvida pela então Sun Microsystems (empresa que posteriormente foi adquirida pela Oracle) em cooperação com empresas de tecnologias de voz e define uma interface de *software* que permite desenvolvedores tirarem vantagem das tecnologias de voz tanto para computação empresarial e pessoal.

A *Java Speech API* define um padrão de interface de *software* multi-plataforma, fácil de usar e que, em sua época, foi o estado da arte na tecnologia de voz. Duas tecnologias principais são suportadas pela JSAPI: reconhecimento e síntese de voz. A Java Speech API foi desenvolvido por meio de um processo de desenvolvimento aberto. Com o envolvimento ativo de companhias líderes em tecnologias de voz, desenvolvedores de *software*, sob meses de revisão pública e atendendo a diversas sugestões, a especificação atingiu um alto grau de excelência técnica.

Os principais objetivos da Java Speech API incluem: prover suporte a sintetizadores de voz e reconhecedores de voz para comando e controle; prover uma interface multi-plataforma robusta para síntese e reconhecimento de voz; permitir acesso ao estado da arte em tecnologia de voz; fornecer suporte à integração com outras funcionalidades da plataforma Java, incluindo *Java Media API*; ser simples, compacto e fácil de aprender.

A *Java Speech API* oferece portabilidade, um ambiente compacto e poderoso, suporte à rede e segurança. Quanto à portabilidade, a linguagem de programação Java, as APIs e a máquina virtual Java estão disponíveis para uma ampla variedade de plataformas de *hardware* e sistemas operacionais além de ser suportado pelo pela grande maioria dos navegadores Web, no que diz respeito ao ambiente compacto e poderoso: a plataforma Java provê aos desenvolvedores uma linguagem poderosa, orientada a objeto, com *garbage collector* (coletor de lixo), que permite um rápido desenvolvimento e maior confiabilidade (alto nível de tolerância a falhas). Por fim, no tocante ao suporte a rede e segurança, existente desde sua concepção, a plataforma Java tem sido voltada para aplicações em rede, com robustez e segurança.

Os recursos de internacionalização oferecidos pela linguagem de programação Java aliado aos caracteres Unicode simplificam o desenvolvimento de aplicações de voz em diversas línguas.

A JSAPI não exige necessidade de *hardware* específico, apenas dispositivos de entrada e saída de áudio comuns.

O Java Speech API em conjunto com o *Java Speech Markup Language* (JSML) fornecem diversas formas para o desenvolvedor de aplicações melhorarem a qualidade do sinal gerado por um sintetizador de voz. O JSML, descrito detalhadamente em uma especificação própria, define marcadores com informações que permitem ao sintetizador melhorar a qualidade da saída resultante, que incluem: marcar o início e o fim de parágrafos e sentenças; especificar pronúncias de qualquer palavra, acrônimo, abreviação ou representações textuais especiais e explicitar controle de pausas, ênfases, entonações, velocidade, volume a fim de melhorar a métrica (SUN MICROSYSTEMS, 1998).

APÊNDICE D: ALGUMAS FERRAMENTAS NATIVAMENTE ACESSÍVEIS VOLTADAS PARA DEFICIENTES VISUAIS

D.1 APL

As linguagens de programação atuais são baseadas em uma interface de linhas de comando interpretadas pelo computador. Tais comandos devem ser corretamente escritos e bem definidos, de tal forma que, caso haja algum erro, seja de sintaxe ou de lógica, o computador será incapaz de compreender as instruções ou as tarefas desejadas não serão realizadas de forma correta. Isto significa que o programador deve memorizar um grande número de instruções. Em resumo: as linguagens de programação atuais são focadas em usuários videntes, pois são fortemente baseadas em interfaces visuais.

Sistemas TTS que leem comandos e variáveis são inadequados para usuários deficientes visuais que desejam programar, sendo o maior problema a verificação de erros. A linguagem APL vem preencher uma lacuna existente entre as linguagens de programação.

APL é uma linguagem de programação com interface baseada em áudio a fim de auxiliar estudantes deficientes visuais na área de desenvolvimento de *software*.

APL foi desenvolvida em Java e se baseia no FreeTTS para a realização da síntese de voz. No APL, o programador não escreve comandos, ele os seleciona a partir de uma lista classificada por categorias, garantindo semântica e sintaxe corretos.

O sistema é composto por duas camadas: *audio interface* (*command list* - lista de comandos a serem escolhidos, como *loop*, condição, entrada, saída, e variáveis, e *query*) e *program logic* (executar, finalizar *loop* ou condição, deletar, salvar comando, verificar próximo passo) e possui dois modos: *programmer* e *running*. A interface é mostrada na Figura D.1.

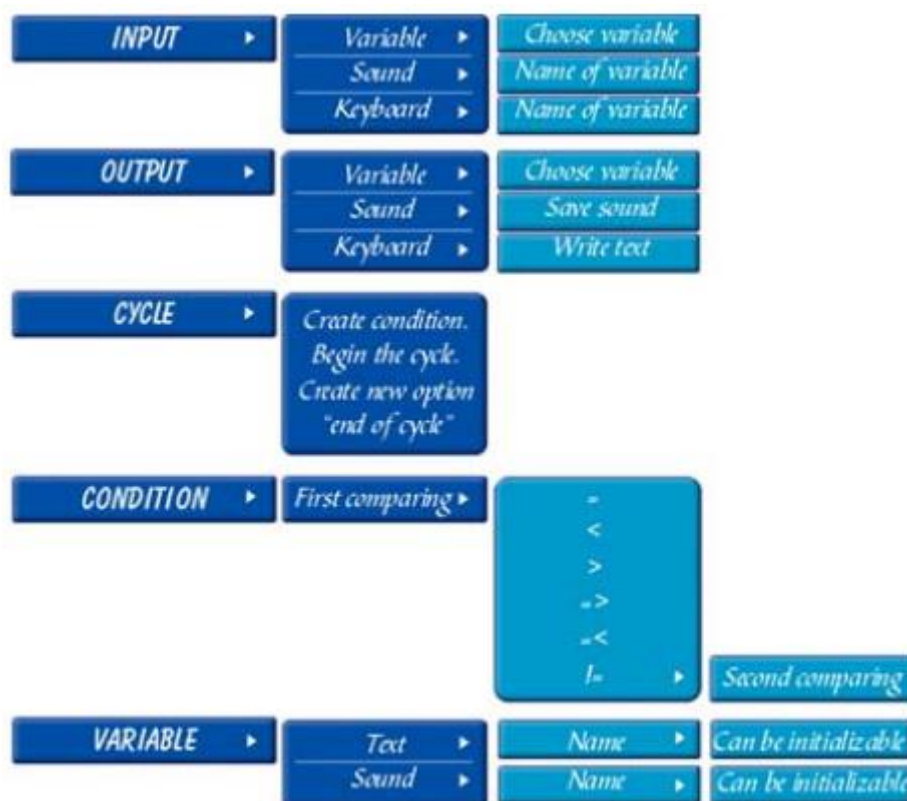



Figura D.1: interface do APL: *Audio Programming Language for Blind Learners*. Fonte: (SÁNCHEZ, 2004).

A linguagem foi desenvolvida para e por deficientes visuais, sendo testado por programadores inexperientes durante e despois do desenvolvimento. Os testes de usabilidade consistiam na proposição de problemas com grau de complexidade crescente e por meio de questionários. Os usuários se mostraram satisfeitos e motivados para interagir com APL, demonstraram interesse e entusiasmo quando programando.

Assim, o APL mostrou ao mercado que é possível construir uma linguagem de programação baseada em áudio que seja capaz de atender usuários deficientes, contribuindo para a inserção destes na área de desenvolvimento de *software* (SÁNCHEZ, 2004).

D.2 Orca

O leitor de tela ORCA é considerado por seus usuários um dos melhores leitores de tela livres para ambiente gráfico, sendo usado pelas distribuições Ubuntu, OpenSUSE, Fedora, Mandriva e Knoppix (COSTA e MONTE, 2012).

O Orca é um leitor de tela escrito em Python para aplicativos gráficos e usa a biblioteca GTK2 e Interface de Provedor de Serviço de Tecnologia Assistiva (AT-SPI,

na abreviação em inglês). O Orca envia rótulos de texto de menus, botões e áreas de texto misto, como o painel principal do navegador web, para um dispositivo Braille e sintetizador de voz. O Orca ainda possui um recurso de ampliador de telas, embora, em alguns teste, não tenha funcionado de forma confiável.

O Orca possibilita o trabalho com o OpenOffice.org 2.3 e versões posteriores com áudio e Braille, contado que o usuário conheça todas as abreviações de teclado necessárias para ativar funções normalmente selecionados com o mouse. O Orca não apenas lê o texto "visual" aparente, mas também oferece dicas e metainformações como família e renderização da fonte, tipos de elementos de formulário e assim por diante. Apesar de estar sendo desenvolvido primariamente para o ambiente GNOME, ele funciona bem com todos os gerenciadores de janela contanto que o aplicativo individual suporte o AT-SPI. Estes incluem o Firefox, OpenOffice.org, Pidgin e até, parcialmente, Gimp.

D.3 Speech Synthesis Markup Language

Em (WALKER et. al. 2001) é possível encontrar uma especificação para linguagem de marcação de texto baseada em XML a fim de possibilitar a interação via voz com sites da internet, denominada SSML (*Speech Syntehsis Markup Language*). Seu projeto é baseado nos seguintes conceitos: consistência, interoperabilidade, generalidade, internacionalização, facilidade de implementação. O SSML apresenta *tags* para definição de línguas, definição de parágrafos e sentenças, personalização de pronúncias, definição de fonemas, configuração do tipo de voz, prosódia, ênfase e inserção de arquivos de áudio.

D.4 VoiceProxy e projeto NatalNet

O Projeto NatalNet (www.natalnet.br) prevê a implementação de um sistema leitor de páginas HTML, cujo objetivo é sintetizar áudio a partir do processamento de páginas HTML. Uma vez pronto, o sistema permitirá que deficientes visuais naveguem através da internet escutando o conteúdo das páginas. VoiceProxy é um sistema em desenvolvimento no contexto deste projeto (SANTOS).

D.5 XLupa

O XLupa é uma lente de aumento (ampliador de tela) digital inteligente para pessoas portadoras de deficiência, particularmente, pessoas com baixa visão. Trata-se de um projeto em conformidade com a filosofia de *software* livre.

O desenvolvimento do XLupa justifica-se por sua natureza inclusiva, digital e, portanto, social.

O XLupa é um *software* desenvolvido em Java, a fim de tirar proveito da portabilidade e produtividade disponibilizadas pela linguagem, e que se encontra em desenvolvimento desde o final de 2004, por pesquisadores do Núcleo de Inovações Tecnológicas (NIT) e do Programa Institucional de Ações Relativas às Pessoas com Necessidades Especiais (PEE), ambos vinculados à UNIOESTE, em parceria com a Secretaria Estadual de Educação do Paraná – CETE / SEED / PR, a Associação de Deficientes Visuais – ACADEVI, o Centro de Atendimento Especializado à Criança – CEACRI e o Centro de Apoio Pedagógico à Pessoa com Deficiência Visual – CAP (BIDARRA, 2005).

ANEXO A: CHEKLIST DE ACESSIBILIDADE PARA *SOFTWARE* IBM – VERSÃO 3.6

Tabela AN1.: Checklist de acessibilidade para *Software* IBM - Versão 3.6

1	**Acesso ao Teclado**	**São Não Planejado N/A**	**Comentários**
1.1	Fornecer equivalência no teclado para todas as ações.		
1.2	Não interferir nas funcionalidades na acessibilidade do teclado incorporadas pelo sistema operacional.		
2	**Informações do Objeto**	**São Não Planejado N/A**	**Comentários**
2.1	Fornecer um indicador de foco visual que se move entre os objetos interativos conforme o foco de entrada vai mudando. Este indicador de foco deve ser programaticamente exposto pela tecnologia assistiva.		
2.2	Fornecer informação semântica sobre objetos de interface do usuário. Quando uma imagem representa m elemento do programa, a informação veiculada pela imagem também deve estar disponível no texto.		
2.3	Associar rótulos com controles, objetos, ícones e imagens. Se uma imagem é usada para identificar os elementos programáticos, o significado da imagem deve ser consistente em todo aplicativo.		
2.4	Quando formulários eletrônicos são utilizado, deve permitir que as pessoas que utilizam a tecnologia assistiva para acessar as informações, elementos de campo e funcionalidade necessária para o preenchimento e envio do formulário, incluindo todas as direções e sugestões.		
3	**Sons e Multimídia**	**São Não Planejado N/A**	**Comentários**
3.1	Fornecer uma opção de sinalização visual para todos os alertas de áudio.		
3.2	Fornecer alternativas acessíveis para áudio e vídeo significativos		
3.3	Fornecer uma opção para ajuste de volume.		
4	**Tela**	**São Não Planejado N/A**	**Comentários**
4.1	Fornecer texto através de sistema padrão de chamada de funções ou através de uma API que suporta a interação com tecnologia assistiva.		
4.2	Uso da cor como um acessório e não como uma única forma de transmitir informações ou indicar uma ação.		
4.3	Suporte a configurações do sistema para alto contraste para todos os controles de interface do usuário e área de conteúdo do cliente.		
4.4	Quando a personalização de cores é suportada, fornecer uma variedade de seleções de cores capazes de produzir uma variedade de níveis de contraste.		
4.5	Herdar configurações do sistema para a fonte, tamanho e cor para todos os controles de interface do usuário.		
4.6	Fornecer uma opção para exibir uma animação em modo de apresentação não-animada.		

5	**Tempo de Resposta**	**São Não Planejado N/A**	**Comentários**
5.1	Fornecer uma opção para ajustar o tempo de resposta de instruções cronometradas ou permitir persistir as instruções.		
5.2	Não usar sinalização ou textos brilhantes, objetos, ou outros objetos tendo brilho com frequência superior a 2HZ e inferior a 55Hz.		

Observação: Não contempla aplicações Web.

Fonte: (SANTOS, 2010).

ANEXO B: QUESTIONÁRIO DE TESTE DE QUALIDADE

UNIVERSIDADE FEDERAL DO CEARÁ

CENTRO DE TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

QUESTIONÁRIO DE TESTE DE QUALIDADE

Aluno: Eng. Nícolas de Araújo Moreira

Orientador: Prof. Dr. Paulo Cesar Cortez

O presente questionário visa coletar resultados qualitativos a respeito do projeto desenvolvido a partir de opiniões, depoimentos e sugestões emitidos por voluntários deficientes visuais para fins de teste e validação.

1. NATURALIDADE DA FALA

A. É produzido por voz humana pré-gravada ou sintetizada por computador?

( ) Voz humana pré-gravada ( ) Sintetizada por computador

B. Qualidade da Voz

[1] Muito Ruim [2] Ruim [3] Razoável [4] Bom [5] Excelente

2. TESTE DE INTELIGIBILIDADE

A respeito das frases a serem sintetizadas:

Olá, seja bem vinda ao projeto LESC Vox. Obrigada por usar o nosso sistema.

Seja bem-vindo ao projeto de acessibilidade "Ver com os ouvidos"! O que você gostaria de fazer?

Quantas palavras não foram entendidas ou foram entendidas de forma errada? _____

3. TESTE DE USABILIDADE

(1) Abrir aplicação de chat e interagir com outro usuário. (2) Abrir editor de texto, digitar mensagem "Isto é um teste do editor de texto", salvar, fechar a aplicação e abrir o arquivo salvo.

4. CONSIDERAÇÕES GERAIS

5. SUGESTÕES DE MELHORIAS